1960

A review of techniques in transportation research

Crane, John W.

Monterey, California: U.S. Naval Postgraduate School

http://hdl.handle.net/10945/12990

A REVIEW OF TECHNIQUES IN TRANSPORTATION RESEARCH

JOHN W. CRANE

A REVIEW OF TECHNIQUES

IN TRANSPORTATION RESEARCH

*****

John W. Crane

A REVIEW OF TECHNIQUES

IN TRANSPORTATION RESEARCH

by

John W. Crane

Lieutenant Commander, United States Navy

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE

United States Naval Postgraduate School
Monterey, California

1 9 6 0

A REVIEW OF TECHNIQUES

IN TRANSPORTATION RESEARCH

by

John W. Crane

This work is accepted as fulfilling

the thesis requirements for the degree of

MASTER OF SCIENCE

from the

United States Naval Postgraduate School

ABSTRACT

Although organized transportation systems are very old indeed, active research into their operations is relatively new. Some of the problems encountered in transportation research are discussed along with methods for their solution. The Hitchcock distribution problem is discussed and a sample problem solved using three different methods. The usefulness of graph theory in studying transportation problems is pointed out. Examples are then given of its use in a maximal network flow problem and in the problem of minimizing equipment requirements to meet a fixed schedule. Finally, some miscellaneous techniques are discussed.

The writer wishes to express his appreciation for the guidance and encouragement given him by Professor Thomas E. Oberbeck of the U. S. Naval Postgraduate School, for the helpful suggestions of the second reader, Professor H. C. Ayers, and for the careful clerical assistance provided by Mrs. Norma A. Stevens.

TABLE OF CONTENTS

INTRODUCTION

In the past decade research into transportation problems has been greatly intensified. Much of this research has involved the application of the principles and techniques from a wide variety of mathematical disciplines. In this thesis we shall summarize some of the existing techniques and give examples of their applicability.

Most of the research we have found reported has as its object the optimization of some feature of a transportation activity. Some examples are: minimizing costs, relieving congestion, minimizing equipment requirements, establishing economical maintenance schedules, minimizing loading and unloading times. Usually the methods reported are directed at solving a problem of a particular transportation activity such as a railroad or an airline. But much of the research has been sufficiently general to be widely applicable. The tendency to generalize models is prevalent. For example, a researcher may seek to minimize the number of "units" required to maintain a fixed schedule. "Units" may then be interpreted by a railroad as boxcars or trains; by an airline, as airplanes; and by a shipping company, as tankers or freighters.

Transportation systems of any appreciable size are extremely complex entities and may defy analysis in many respects. Researchers turn to mathematical models of the systems they wish to analyze and build into the model the degree of detail or generality necessary to give significant results. This practice has resulted in the application of a great many mathematical techniques to problems in transportation. The techniques and disciplines which have been used include: linear programming, transhipment and assignment techniques, non-linear and dynamic programming, queuing, or waiting line theory, graph theory, and the Monte Carlo method.

In our research for this paper we have found a great deal of material on transportation type problems in the following publications: The Journal of the Operations Research Society of America, Management Science, Naval Research Logistics Quarterly, and others listed in bibliographies at the end of each chapter. Groups we have found to be particularly active in transportation research include: The Rand Corporation, Santa Monica, California; the George Washington University Logistics Research Project; The Cargo-Handling Research Project at the University of California at Los Angeles; The Management Sciences Research Group, Purdue University; to name only a few.

Professor F. Harary of the University of Michigan has done a considerable amount of research in graph theory, which is a directly supporting discipline. He is co-author of a new book on this subject which is due to appear in 1960. This book will be the first full treatment of this subject in English.

As an indication of the direction in which current research is tending, we point to Rand report R-351 which gives, in abstract form, the proceedings of "The Rand Symposium on Mathematical Programming". In his address to the opening session of this symposium, G. B. Dantzig cited the five following theoretical areas as those in which future developments are expected:

(1) Special Structures
(2) Discrete Programming
(3) Network Theory
(4) Non-linear Programming
(5) Uncertainty

It is anticipated that transportation theory will be one of a great many beneficiaries of developments in these areas.

In arranging various types of transportation problems into groups for the ensuing discussion we were, of necessity, a bit arbitrary. We have presented in detail only a few of the more widely used techniques and have made reference to others. We have tried to present a reasonably complete bibliography at the end of each chapter.

In Chapter I we have stated and discussed some methods for the solution of the Hitchcock distribution problem, both with and without capacity limitations. These are essentially linear programming problems. In Chapter II we have indicated the applicability of linear graph theory to transportation problems and general network problems. In Chapters III and IV we have we have discussed in some detail two problems that illustrate the applicability of graph theory to transportation problems. In Chapter V we have cited several examples of other techniques and the problems to which they have been applied. Specifically, we have discussed queuing theory, simulation, and straightforward engineering analysis including the difficult and time consuming task of data gathering.

CHAPTER I

THE TRANSPORTATION PROBLEM IN LINEAR PROGRAMMING

1. The Hitchcock Distribution Problem.

The problem to be discussed in this chapter was originally stated in 1941 by F. L. Hitchcock [1][1] and was further discussed and elaborated by T. C. Koopmans [2]. The problem may be stated as follows: Certain amounts, a_i $(i = 1,\ldots,m)$, of a homogeneous product are available at each of m origins, and certain amounts b_j $(j = 1,\ldots,n)$ are required at each of n destinations. The cost, c_{ij}, of shipping a unit amount from the i^{th} origin to the j^{th} destination is known for all i and j. We will further stipulate $\sum a_i = \sum b_j$. The problem is to select the amounts x_{ij} to be shipped form each origin to each destination so that the total shipping cost is a minimum. The amounts x_{ij} must be non-negative, $x_{ij} \geq 0$.

Algebraically, we state the problem in the following form:

Given: (1) Two sets of constants a_i $i = 1,\ldots,m$

and b_j $j = 1,\ldots,n$

such that

$$\sum a_i = \sum b_j, \text{ and}$$

(2) A matrix of constants $C = [c_{ij}]$ $i = 1,\ldots,m$

Find: A set of non-negative unknowns

$x_{ij} \geq 0$, satisfying

$$\text{(a)} \sum_i x_{ij} = b_j \qquad j = 1,\ldots,n$$

$$\text{(b)} \sum_j x_{ij} = a_i \qquad i = 1,\ldots,m$$

1. Numbers in brackets refer to references cited in the Bibliography

(c) $\sum_i \sum_j c_{ij} \; x_{ij}$ = Minimum

Linear programming problems which may be stated in the foregoing form have come to be called "the transportation problem" even though they do not all deal with transportation. Examples of such problems are the personnel assignment problem, the contract - award problem, [3] [4] the Traveling-Salesman problem [10] , and the Caterer problem [11] .

G. B. Dantzig [5] was first to formulate this problem as a linear - programming problem, and he gave an algorithm for its solution based on his Simplex method [6] . The transportation algorithm is considerably easier to apply than the general simplex method in that it does not require the inversion of matrices. The algorithm is discussed in more detail and a sample problem worked out in Appendix A.

2. The Assignment Technique.

As was noted above, a special case of this "transportation problem" is the n x n assignment problem. In this case all $a_i = b_j = 1$, and all x_{ij} are either one or zero. The problem may be stated as follows: determine the optimum assignment of <u>n</u> persons to <u>n</u> jobs, given a matrix of constants $[r_{ij}]$ which specify, in appropriate units, the worth or performance rating of the i^{th} individual in the j^{th} job. The most straightforward approach is to consider each of the n! possible arrangements. But, for only moderately large n, this becomes a hopeless task. What is required is a procedure for finding an optimum assignment (there may be more than one) with a reasonable amount of effort. Such a procedure was provided by H. W. Kuhn [3] in his "Hungarian Method".

Kuhn's algorithm has been modified and extended to the m x n

transportation problem by L. R. Ford, Jr. and D. R. Fulkerson [7], and by J. Munkres [8]. The algorithms so developed may be described as procedures that make maximal use of minimal cost routes. Each assigns quotas only to the least expensive feasible routes until the assignment is complete. The resulting assignment is one which gives minimum cost. It is not necessarily unique. These algorithms are also discussed more fully in Appendix A and a problem is solved using each in turn.

3. The Transhipment Technique.

An interesting variation on the $m \times n$ transportation problem (m sources, n destinations) is the transhipment problem described by Alex Orden [9] In the original transportation problem each point acted as a shipper only or as a receiver only, and the routes were considered to be direct from each shipper to each receiver. In his extension of the problem, Orden has permitted shipments to go via any sequence of points rather than being restricted to direct connections from one of the origins to one of the destinations. His method of solution involves converting the extended problem to the original problem by a rather simple device. Each point is treated as a pair of points, one acting as a shipper and one as a receiver. The unit cost of shipment from a point considered as a shipper to the same point considered as a receiver is set equal to zero. All other c_{ij} $(i, j = 1, \ldots, m + n, \ i \neq j)$ are assumed known and greater than zero. For computational purposes large stockpiles are assumed to exist at each shipping point. The excess of stockpiles over actual amounts shipped appear as shipments from a point to itself at zero cost and are removed from the final result. With this mechanism, the $m \times n$ problem with transhipment is converted to an $M \times M$ problem

without transhipment (a Hitchcock problem), where $M = m + n$. The problem is solved in this form and, in the non-degenerate* case, $2M - 1$ routes have positive quotas assigned. It happens that in minimum-cost solutions all x_{ii} are positive, but they are removed leaving $M - 1 = m+n-1$ actual shipments.

4. Integer Solutions.

It is worth noting here that if the Hitchcock problem is posed in integers (i.e. each a_i and b_j is a positive integer) then there is at least one minimizing solution in which each x_{ij} is either a positive integer or zero [4]. In this case the whole problem can be solved in integers by introducing trial, feasible solutions consisting of integers and then changing these only by integral amounts. Very recently, Gomory [13] has developed a "method of integer forms" for solving a general linear programming problem with the additional constraint that the variables in the solution be integers. Dantzig [14] has included this procedure in the term "discrete programming" and forsees in its further development promise of solving "all kinds of combinatorial, non-linear, non-convex problems...areas where classical mathematics has been weakest".

5. The Capacitated Problem.

In the classic Hitchcock problem the upper bound on the amount of commodity assigned to any route is determined by the amounts available at the sources, a_i's, and the amounts required at the destinations, b_j's. The routes themselves are not regarded as limited in their capacities. One can imagine practical situations wherein the actual capacities of the routes may be limiting factors. For example, suppose it is desired to

*See Appendix A for a discussion of degeneracy.

transport certain known quantities of commodity from several sources to several destinations each day, or month, and we have available certain railway routes with specified capacities; then, the amount of the commodity shipped on route i-j, in the specified time interval, cannot exceed the capacity of that route.

Our introduction of capacities has brought time into the problem, at least to a limited degree. The natural specification of capacity is a rate. Consider a problem similar to the Hitchcock problem, except suppose the a_i's represent quantities to be shipped in a certain time interval, say a month, and the b_j's are amounts required at the destinations per month. Further, let us specify the capacity of each route i-j in units of commodity per month. We have then the so - called capacitated Hitchcock problem. This problem is similar to the uncapacitated problem, except that we envision it as being repeatedly applicable within time intervals of specified length. Basically, it is static in that we have introduced no truly dynamic variables such as instantaneous flow rates or transit times. And we have supposed the given constants-a_i's, b_j's, and the capacities- do not change with time.

We state the problem formally.

Given:

(a) Two sets of constants

$a_i \qquad i = 1,\ldots,m$

$b_j \qquad j = 1,\ldots,n$

Such that $\sum a_i = \sum b_j = k$

(b) A matrix $C = \left[c_{ij}\right]$

(c) another matrix $D = \left[d_{ij}\right]$

where c_{ij} represents the capacity of route ij and

d_{ij} represents the cost of transporting a unit amount from source i to destination j.

Find $X = \left[x_{ij} \right]$

such that

$$(1) \quad \sum_i x_{ij} = b_j$$

$$(2) \quad \sum_j x_{ij} = a_i$$

$$(3) \quad 0 \leq x_{ij} \leq c_{ij}$$

$$(4) \quad \sum_i \sum_j x_{ij} d_{ij} = \text{minimum}$$

Whereas the uncapacitated problem always has a feasible solution [4], the capacitated problem may not. The problem is feasible if, and only if, the maximal flow of the associated network is at least equal to the total amount to be shipped, K. A discussion of networks and their maximal flows will follow in Chapters II and III.

Ford and Fulkerson have also provided a method for solving the capacitated problem in their paper, "A Primal - Dual Algorithm for the Capacitated Hitchcock Problem" [15]. As their title implies, these authors define a dual problem involving dual variables α_i, β_j and γ_{ij}. These are used to distinguish certain routes, and, in terms of them, a restricted primal problem is solved. From this solution, new dual variables are defined and the restricted primal problem is again solved, this time using the new dual variables. In each solution of the restricted primal problem, a set of positive x_{ij} is determined. When the sum of these x_{ij} is equal to the total amount to be shipped, K, the x_{ij} so determined provide a solution to the original problem.

During each iteration, in determining the solution to the restricted

primal problem, the authors call upon a special flow algorithm developed by them in a previous paper [16]. The mechanics of the solution are similar to those described for the uncapacitated problem and will not be described in further detail here.

In a subsequent paper [17], Fulkerson showed that an upper bound on the number of iterations or "labelings" required to solve the transportation problem could be determined. These upper bounds were: (a) for an $n \times n$ optimal assignment problem, $n(n+1)$; (b) for the transhipment problem, $K(n+1)$; (c) for a Hitchcock - Koopmans problem $K\left[\min(m,n)+1\right]$.

The literature contains numerous other examples of applications of linear and non-linear programming techniques to transportation problems. Several such papers are cited in the bibliography without special reference to them in the text. These are intended to be representative, rather than exhaustive; to attempt an exhaustive listing would be presumptuous, indeed.

BIBLIOGRAPHY

1. Hitchcock, F. L., "The Distribution of a Product from Several Sources to Numerous Locations" Journal of Mathematics and Physics Volume 20 pp 224-230 (1941).

2. Koopmans, T. C., "Optimum Utilization of the Transportation System" Econometrica Volume 17, Supplement, 1949.

3. Kuhn, H. W., "The Hungarian Method for the Assignment Problem", Naval Research Logistics Quarterly, Volume 2 Numbers 1 and 2, March-June 1955.

4. Gass, Saul I., Linear Programming Methods and Applications McGraw-Hill Book Company, Inc., New York, 1958.

5. Dantzig, G. B., "Application of the Simplex Method to a Transportation Problem", Activity Analysis of Production and Allocation, Cowles Commission Monograph 13, John Wiley & Sons, Inc., New York, 1951.

6. Dantzig, G. B., "Maximization of a Linear Function of Variables Subject to Linear Inequalities", Activity Analysis of Production and Allocation, Cowles Commission Monograph 13, John Wiley & Sons Inc., New York, 1951.

7. Ford, L. R., Jr., and Fulkerson, D. R., "Solving the Transportation Problem", Management Science, Volume 3, Number 1, October 1956.

8. Munkres, James, "Algorithms for the Assigment of Transportation Problems", Journal of the Society for Industrial and Applied Mathematics, Volume 5, Number 1, March 1957.

9. Orden, Alex, "The Transhipment Problem", Management Science, Volume 2, Number 3, April 1956.

10. Flood, Merril M., "Traveling - Salesman Problem", Journal of Operations Research Society, Volume 4, Number 1, 1956.

11. Prager, William, "On the Caterer Problem", Management Science, Volume 3, Number 1, October 1956.

12. Flood, Merril M., "Application of Transportation Theory to Scheduling a Military Tanker Fleet", Journal of the Operations Research Society of America, Volume 2, 1954, pp 150-162.

13. Gomory, Ralph E., "The Method of Integer Forms", (Abstract) The Rand Symposium on Mathematical Programming, Philip Wolfe, Editor. The Rand Corporation, Santa Monica, California, 1960.

14. Dantzig, G. B., "New Directions in Mathematical Programming", The Rand Symposium on Mathematical Programming, Philip Wolfe, Editor, The Rand Corporation, Santa Monica, California, 1960.

15. Ford, L. R., Jr., and Fulkerson, D. R., "A Primal - Dual Algorithm for the Capacitated Hitchcock Problem", Naval Research Logistics Quarterly, Volume 4, Number 1, 1957.

16. Ford, L. R., Jr., and Fulkerson, D. R., "A Simple Algorithm for Finding Maximum Network Flows and an Application to the Hitchcock Problem". The Rand Corporation Paper, pp 743, September 26, 1955.

17. Fulkerson, D. R., "Bounds on Primal - Dual Computation for Transportation Problems". The Rand Corporation RM 2178, May 21, 1958.

18. Ferguson, Allen R., and Dantzig, George B., "The Allocation of Aircraft to Routes - An Example of Linear Programming Under Uncertain Demand", Management Science, Volume 3, Number 1, October 1956.

19. Beale, E. M. L., "An Algorithm for Solving the Transportation Problem when the Shipping Cost over Each Route is Convex", Naval Logistics Research Quarterly, Volume 6, Number 1, March 1959.

20. Evans, G. W., II, "A Transportation and Production Model", Naval Research Logistics Quarterly, Volume 5, Number 2, June 1958.

21. Heller, I. "Constraint Matrices of Transportation Type Problems", Naval Research Logistics Quarterly, Volume 4, Number 1, March 1957.

22. Manne, Alan S., "Allocating MATS Equipment With the Aid of Linear Programming", The Rand Corporation, RM -1612.

23. McCloskey, J. F., and Hannsmann, Fred, "An Analysis of Stewardess Requirements and Scheduling for a Major Domestic Airline", Naval Research Logistics Quarterly, Volume 4, Number 3, September 1957.

CHAPTER II

TRANSPORTATION NETWORKS AND GRAPH THEORY

1. Discussion.

In the study of transportation systems it is often desirable to utilize network diagrams. The usual approach is to let the nodes of the network represent origins, destinations, or junction points, and to connect these nodes with lines representing the available routes. The description, representation, and use of such diagrams has been greatly facilitated by the use of the mathematical theory of linear graphs An early example of the use of graph theory in connection with transportation problems was in a paper by T. C. Koopmans and S. Reiter [1] published in 1951. Much of the work had been done earlier during World War II by the former author.

The time - lag in the application of graph theory to transportation problems is emphasized by quoting from the above mentioned paper.

> "The cultrual lag of economic thought in the application of mathematical methods is strikingly illustrated by the fact that linear graphs are making their entrance into transportation theory just about a century after they were first studied in relation to electrical networks, although organized transportation systems are much older than the study of electricity."

Perhaps some of this delay is due to the fact that, until very recently, graph theory has been somewhat neglected by mathematicians themselves. Few articles on the subject appear in mathematical journals.

Until 1958, there was only one book devoted entirely to graph theory - D. Konig's Theorie der Endlichen und Unendlichen Graphen [2], first published in Leipzig in 1936 and reprinted by Chelsea in New York in 1950. The second book on the subject, by Claude Berge [3], was published in

France in 1958.

Professor F. Harary, of the University of Michigan, has written quite extensively on graph theory and, with R. Z. Norman, is writing a book due to appear in 1960. [5]

Because of its wide applicability and comparative simplicity, the two-terminal [1] network is used in many transportation problems. Normally, flow in such a network is unidirectional from a source to a sink.

This theory has been especially useful in determining the maximal flow which a network will sustain. Examples of its use in this connection are included in Chapter III.

The similarities between communications and transportation network problems have been recognized and pointed out by R. E. Kalaba and M. L. Juncosa [4], among others. This similarity is not limited to their common tie with graph theory but extends also to queuing theory, simulation devices, linear programming, dynamic programming, and Boolean algebra.

2. Representations.

We shall exclude from our discussion graphs containing arcs which join individual nodes to themselves.

A natural representation of a network is a diagram in which points represent nodes, and lines represent the routes between them. Elsewhere, (Appendix C) we have defined a network to be a (connected) graph G, together with the capacities of its individual arcs. We shall illustrate here a one-origin, one terminal planar network with source a and sink b.

1. For definitions of this and other technical terms, see Appendix C.

The graph is not oriented.

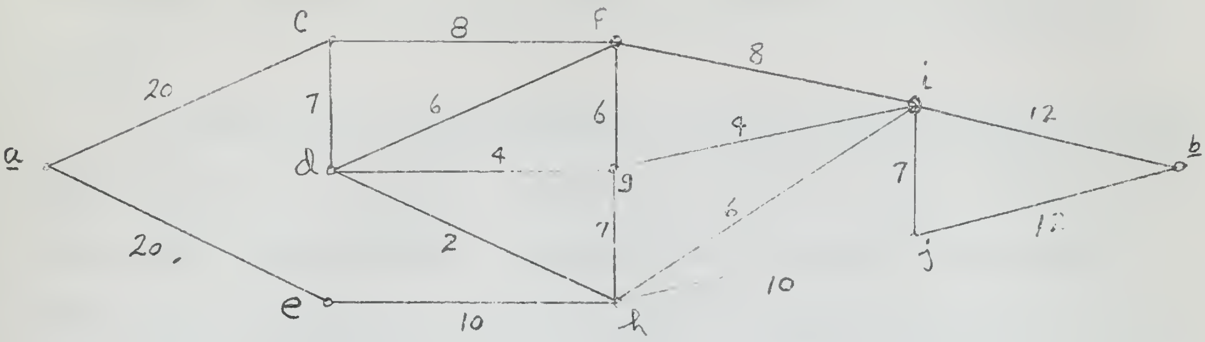


Figure 2-1

This network could also be represented by a symmetric matrix $L = [\ell_{ij}]$ in which ℓ_{ij} is the capacity of arc ij if the arc ij is included in the graph, and is zero otherwise. For our example the applicable matrix would be the one below. Except on the main diagonal, zeros are not written but are implied by a blank space.

	a	b	c	d	e	f	g	h	i	j
a	0		20		20					
b		0							12	[illegible]
				7		8				
			7	0		6	4	2		
	20				[illegible]			10		
			8	6			6		8	
				4		6		7	4	
				2	10		7		[illegible]	10
		[illegible]				[illegible]			0	7
		2						10	7	0

Figure 2-2

In Chapters III and IV we shall use the language of the theory of graphs. For representations of the graphs which we study diagrams such as figure 2-1 rather than the matrix representation, figure 2-2, will be used.

It is desirable to emphasize that we have pointed out only one field for the application of graph theory. In addition to their usefullness in the study of transportation, communications, and electrical networks, graphs have been used in many other fields. They are useful in describing Markov chains, for example. As another example Professor Harary has made use of them in the study of the behavioral sciences [6], [7]. Much of his work in graph theory has been done in conjunction with the Research Center for Group Dynamics, University of Michigan, and the Social Science Research, Bell Telephone Laboratories.

BIBLIOGRAPHY

1. Koopmans, T. C. and Reiter, S., "A Model of Transportation" Activity Analysis of Production and Allocation, Cowles Commission for Research in Economics, T. C. Koopmans (editor). John Wiley & Sons, Inc., New York, 1951.

2. Konig, D., Theorie der Endlichen und Unendlichen Graphen, Leipzig, 1936, reprinted Chelsea Publications Inc., New York, 1950.

3. Berge, Claude, Theorie des Graphes et Ses Applications, Dunod, Paris, France, 1958.

4. Kalaba, R. E., and Juncosa, M. L., "Communication - Transportation Networks (Abstract)", Naval Research Logistics Quarterly, Volume 4, Number 3, September 1957.

5. Harary, F. and Norman, R. Z., The Theory of Graphs, due to appear in 1960.

6. Harary, F. and Norman, R. Z. "Graph Theory as a Mathematical Model in Social Science", Institute for Social Research, Ann Arbor, 1953.

7. Harary, F., Norman, R. Z., and Cartwright, D., Introduction to Digraph Theory for Social Scientists, in process of publication.

CHAPTER III

MAXIMAL NETWORK FLOW

1. General

As we mentioned in Chapter II, transportation systems are often represented by network models. The nodes may represent simple transfer points such as the pick up point for the hook in a cargo-handling operation; or, they may represent highly complex entities such as a seaport or a complete railway operating division. What the nodes represent depends upon the model under consideration and the degree of generality or detail under which it is being studied. These nodes are joined by arcs which may have an associated traffic-carrying capacity. This collection of points and arcs with their capacities will be referred to as a network - (see Appendix C).

One question of interest is that of determining the maximal steady-state flow through such a capacitated network. We shall describe in this chapter two procedures for determining maximal flow, one a flooding technique due to Alexander W. Boldyreff [1] and the other the minimal cut theorem due to L. R. Ford, Jr., and D. R. Fulkerson [2].

2. The Flooding Technique

Boldyreff applied his technique directly to traffic through a railroad network, and was careful to claim no generality for it. He characterized it as empirical in nature, and left as an open question the possibility of wider use of the flooding technique. It appears feasible to apply it in any situation where a network model will fairly describe the actual situation.

The method employed can be applied to networks with any number of

origins and terminals; however, for clarity the one-origin, one-terminal case is described.

Certain subsets of points are distinguished as the origins, the terminals, and junction points (see Appendix C). We will discuss flow in a railway network assuming capacities in trains per day. By "steady-state" we mean that the flow rate through each arc, in trains per day, is constant from day to day. The steady-state is further clarified in these initial assumptions:

1. At the origins only loaded trains leave and only empty trains arrive.
2. At the terminals only loaded trains arrive, and only empty trains leave.
3. At each junction point the number of loaded (empty) trains arriving is equal to the number of loaded (empty) trains leaving each day.
4. The number of loaded trains leaving all the origins is equal to the number arriving at all the terminals each day.
5. The number of empty trains leaving all the terminals is equal to the number arriving at all the origins each day.

The last two statements are deducible from the first three and are stated explicitly only for emphasis.

The problem is to find a method of assigning the steady-state flow of traffic to each arc of the network, not exceeding the capacity of the arc, which will maximize the total flow of loaded (empty) trains from the origins (terminals) to the terminals (origins), and satisfy

the five assumptions stated above.

In the problem as stated, the steady-state consists of simultaneous movement through the network of loaded trains from the origins to the terminals and empty trains in the opposite direction. These two movements must necessarily be at the same rate. By imagining a network with arc capacities just one-half of those of the original network, we can look only at the flow of loaded trains from origins to terminals and reduce the problem to the unidirectional case. By unidirectional flow we mean that the flow in any one arc is in one direction only. In this discussion the flow considered will be generally directed from the origins to the terminals.

For this simplified problem the steady-state conditions are contained in the following statements:

1. At the origins all trains leave, none arrive.
2. At the terminals, all trains arrive, none depart.
3. At each junction point the number of trains coming in is equal to the number going out.
4. The number of trains leaving all origins must equal the number arriving at all terminals.
5. The traffic flow through each arc cannot exceed the capacity of that arc.

By defining two networks to be equivalent if the maximal flows through them are equal, we can readily reduce a network with many origins and many terminals to an equivalent network with a single origin and a single terminal.

We shall describe the flooding technique utilizing a single-origin,

single-terminal network. The procedure is as follows:

Starting at the origin, assign sufficient traffic flows to all arcs leaving the origin to saturate them. This gives the maximum number of trains arriving at junction points one arc removed from the the origin.

View this set of junction points as new origins and, starting with the one subject to the greatest capacity constraint, schedule trains whenever possible in the following order:

1. 'Forward' - to new junction points through the outgoing arcs
2. 'Laterally' - to other points of the set.
3. 'Bottlenecked' - if trains are left over after steps 1 and 2; i.e. if all outgoing and lateral arcs are saturated.

At such a set of junction points there may be arbitrary decisions regarding which trains to 'bottleneck'. The guiding principle is to move forward as many trains as possible, and to maintain the greatest flexibility for the remaining network.

Continue the above procedure until the scheduled flow covers the complete network and reaches the terminal.

Eliminate bottlenecks by returning all excess trains to the origin. The validity of the solution can be checked by inspection. If a maximal flow has been achieved, there will be no continuous unsaturated path extending from the origin to the terminal. (That is there will be no chain connecting origin and terminal that does not contain at least one saturated arc). If this criterion is not met, the inspection will have revealed the unsaturated chains. Flow in these chains can be increased, giving maximal flow. As will be seen, the observation concerning saturated arcs is a point of similarity between this procedure and the

minimal cut procedure to be discussed.

In this paper, Boldyreff also gives some procedures for simplifying complicated networks. However, as he points out, the networks to which these procedures can be applied are rather more the exception than the rule, and the applications are tedious. The author recommends a straightforward application of the flooding technique.

A sample problem is solved by this procedure in Appendix B. The same problem is also solved using Ford and Fulkerson's minimal cut theorem.

3. The Minimal Cut Theorem.

This method was applied to networks, in general, not necessarily rail networks. Although we will not give here a proof of the minimal cut theorem[1], we will need some definitions in order to discuss its applications. See Appendix C.

A <u>graph</u>, G, is a finite, one-dimensional complex, composed of vertices a,b,c,...,e and arcs α (ab), β (ac),..., δ(ce).

An <u>arc</u> α (ab) joins its end vertices a,b it passes through no other vertices of G, and intersects other arcs only in vertices.

A <u>chain</u> is a set of distinct arcs of G which can be arranged as α (ab), β (bc),..., δ(gh) where the vertices a,b,c,...,h are distinct. A chain does not intersect itself, it <u>joins</u> its end vertices <u>a</u> and <u>h</u>. Each arc in G has associated with it a positive number called its <u>capacity</u>. The graph G, together with the capacities or its individual arcs, is called a <u>network</u>. Two vertices of G are distinguished <u>a</u>, the source and <u>b</u> the sink. A <u>chain flow</u> from <u>a</u> to <u>b</u> is a couple (C,k) composed of a

1. For a proof of the thorem, see the second article listed in the bibliography at the end of this chapter.

chain joining a and b and a non-negative number k representing the flow along C form source to sink. A flow in a network is a collection of chain flows which has the property that the sum of the numbers of all chain flows that contain any arc is no greater than the capacity of that arc. If equality holds, the arc is said to be saturated by the flow. A chain is saturated with respect to a flow if it contains a saturated arc. The value of a flow is the sum of all the chain flows which compose it. A disconnecting set D, is a set of arcs which has the property that every chain joining a and b meets the collection. A disconnecting set, no proper subset of which is disconnecting, is called a cut.

The value of a disconnecting set $v(D)$ is the sum of the capacities of its individual members. Thus a disconnecting set of minimal value is automatically a cut.

We state the minimum cut theorem.

Theorem 1. The maximal flow value obtainable in a network is the minimum of $v(D)$ taken over all disconnecting sets D.

This theorem is so intuitively appealing that it hardly seems to require proof at all. However, as examination of reference [2] will show, the proof is quite involved, and for certain networks (for example see [3]) it does not apply. The inapplicability to networks with several sources and corresponding sinks was illustrated also by Ford and Fulkerson [2].

The actual computation procedure for determining the value of the minimal cut is based on a corollary to the minimal cut theorem and an additional theorem which will be stated without proof.

> Corollary: Let A be a collection of arcs of a network N which meets each cut of N in just one arc. If N' is a network obtained from N by adding k to the capacity of each arc of A, then cap $(N') = \text{cap}(N) + k$

Theorem 2. If the graph of G, together with arc ab, is a planar graph [4], there exists a chain joining _a_ and _b_ which meets each cut of N precisely once.

Let T be the chain joining _a_ and _b_ which is topmost in N. T has the property of theorem 2. Impose as large a chain flow as possible (T,k) on this chain, thereby saturating one of its arcs.

By the corollary, subtract k from the capacity of each arc of T. Remove the previously saturated arc whose capacity is now zero. Continue this procedure. Eventually the graph disconnects and a maximal flow has been established.

For an illustrative example, see Appendix B.

In problems of sea-transport, the most restrictive capacity limitations are quite often the cargo-handling capacities of ports at which ships must load and/or unload, rather than on sea routes themselves. This is often true even when the number of ships available is limited. At first, it would appear that this is a problem wherein the capacity restrictions are on the nodes of a graph rather than its arcs. We suggest, however, that this can be reduced to the previous problem by employing a device similar to that used by Orden. See Chapter I. Treat each node representing a port of limited capacity as both a receiver and a shipper and replace the single node by two nodes joined by an arc showing the capacity. We have merely introduced a slight modification to the model to show more detail.

The applicable units of capacity might be troublesome in a shipping network due to the wide variety of ships with which one might deal. There is a great temptation to dismiss this difficulty by introducing some sort of a "standard" or "notional" ship, and for a first approximation this may suffice. However, some ports may well be able to handle,

for example, 3 notional ships per day provided this capacity actually came in three ships near our "standard" size, and be incapable of even docking one very large ship of equivalent capacity.

Thus our model begins to break down and we need to introduce a more realistic one capable of handling the real situation. We would emphasize that the models with which we have dealt to this point are mostly systems models dealing with large-scale inputs and gross effects. They would require inputs representing aggregation of many lesser inputs, and these must often be combined by skillful application of lessons provided only by experience. Simple arithmetic summations rarely suffice. The interpretation and application of the results provided by the model is equally difficult, and requires the same careful judgment.

BIBLIOGRAPHY

1. Boldyreff, Alexander W., "Determination of the Maximal Steady State Flow of Traffic Through a Railroad Network" JORSA Volume 3. Number 4, 1955.

2. Ford, L. R., Jr., and Fulkerson, D. R., "Maximal Flow Through a Network" Canadian Journal of Mathematics, Volume 8, 1956, pp. 399-404.

3. Robacker, J. T., "Concerning Multicommodity Networks", The Rand Corporation, RM-1799, 26 September 1956.

4. Whitney, H., "Non-Separable and Planar Graphs", Transactions of the American Mathematical Society Volume 34, pp. 339-362 (1932).

CHAPTER V

MINIMUM TRANSPORT UNITS TO MAINTAIN A FIXED SCHEDULE

1. General Discussion.

We shall consider in this chapter a problem which is, in a sense, complimentary to the problem of maximal flow previously discussed. We shall be concerned with the determination of the minimum number of transport units necessary to maintain a fixed schedule.

In order to make precise the notion of a schedule, we shall again use the language of graph theory. We envision a connected graph in which a subset of nodes is distinguished as terminals; others represent junction points. By a run we mean the traversing of a certain path from one terminal to another by a single transport unit. Runs originate (depart) and terminate (arrive) at certain specified times within an interval, such as a day or a week. These departures and arrivals are repeated indefinitely in past and succeeding intervals of the same duration. We choose a typical time interval of length π, commencing at some arbitrary time t_o and ending at time $t_1 = t_o + \pi$. A schedule is the specification of the time of each departure and arrival during the time interval π, for each terminal in the system. It will be noted that the duration of a run may be less than, equal to, or greater than our typical interval.

Perhaps the simplest feasible solution of a schedule problem is to assign each unit to repeated round-trips between two terminals. This may involve the movement of a great many empty or "dead-head" units, and is rarely optimum, if other utilization of units is feasible. As can be readily seen, the problem of minimizing transport units to meet a prescribed schedule reduces to the problem of routing empties [1], [2].

We have found several procedures for determing minimum units for fixed-schedule operations, [2], [3], [4], [5]. In this chapter we shall describe, as a representative example, an algorithm by T. E. Bartlett [3] built around a railway schedule.

2. Algorithm for Minimum Transport Units.

Assumptions

(1) Schedules under consideration refer to runs of transport units between terminals. The runs originate or terminate, at specified times, within an interval, of specified duration, and are repeated indefinitely in past and succeeding intervals of the same length.

(2) Each terminal has at least one route connecting it with one or more other terminals. Any terminal may be reached from any other in the system. The system may be represented by a connected graph.

(3) Each terminal has as many "departing" runs as "incoming" runs. This may require transfer of empty or "dead-head" units.

(4) For each run there is a standard or minimal type of transport unit which is employed. Units may be employed on any run for which specifications are met.

The above assumptions are quite reasonable and are usually satisfied by transportation systems operating on schedules.

Additional Assumptions

For the particular algorithm presented, certain additional, restrictive assumptions were made. These were:

(1) Requirements at any terminal for transport units to make up outgoing runs must be obtained from regularly scheduled incoming runs at that terminal.

(2) All transport units are of a single type.

(3) All schedules are met precisely.

(4) No delay beyond the time differential of scheduled arrivals and departures is required for interchange of transport units from one run to another.

Discussion

By the hypothesis concerning the schedule a unit is either on a run or it is "idle" at some terminal awaiting transfer to a run. A schedule which is maintained generates a total running time plus idle time which is equal to the number of units employed multiplied by the length of a typical period. Since the total running time is fixed by the schedule, the number of units employed will be a minimum if, and only if, the total idle time is at a minimum. Therefore, what is required is a pairing of arrivals and departures at each terminal which will minimize total idle time.

Three separate cases are considered.

Case I. All departures in a typical period at a terminal are later than all arrivals during this period.

Case II. Some departure is earlier than some arrivals, but all arrivals and departures within the period can be paired in at least one way so that each departure is later than its paired arrival.

Case III. Up to some time during the period, more departures than arrivals have been scheduled.

In the first case any pairing of departures and arrivals generates the same total idle time. In the second case, pairing of the "first in" with the "first out" is both feasible and yields minimum idle time.

Case III presents a special problem. Since there are, up to a point, more departures than arrivals, there is no feasible pairingof all departures with arrivals within one period. It is necessary to pair one or more

departures with arrivals from a preceeding period and some arrival (s) within the period are left over for pairing with departure(s)in a subsequent period. By extending the sequence both forward and backward sufficiently to pair all departures with arrivals, the pairing problem reduces to that of Case II, and the first-in-first-out procedure gives minimum idle time. However, it is necessary to adopt a consistent association of idle time to a particular period in order not to miscount the total time. The procedure adopted was to associate with the current period that idle time generated by pairing runs of the previous period to runs in the current period as well as the idle time generated by pairing within the period.

Idle time developed in the pairings from the current period to the next period is not charged to the current period, but to the next one.

Results

From the above considerations a total, running plus idle, time is calculated, and is shown to be.

$$T = \pi (\alpha + \sum_j \beta_j)$$

where

π = length of period

α = total runs en route at the beginning (or end) of a period for all terminals

β_j = maximum value of cumulative departures less cumulative arrivals at terminal j during a period.

The minimum number of transport units required is then obtained by dividing T by the length of the period π, and is given as:

$$U = \text{minimum number of transport units} = \alpha + \sum_j \beta_j$$

In concluding his paper, Bartlett pointed out some areas of scheduling problems under study. These are listed below. Some of them, it can be seen, are aimed at removing the restrictive assumptions made earlier. Items listed were:

a. Determination of routings to permit economic maintenance schedules.

b. Determination of routings to optimize the number of units used for the situation in which several classes of equipment may be employed on various routes.

c. Determination of routings to minimize the number of units used and maintain schedules having probabilistic arrival times.

d. Determination of optimum schedules based on cost of transport units, operating costs, and demand distributions.

e. Analysis of the possible application of the concepts in the field of machine scheduling.

The actual problem of allocating units to a schedule in a manner to preserve minimality and, at the same time, meet certain maintenance requirements, was discussed in a subsequent paper by T. E. Bartlett and A. Charnes [5]. Their method of solution involves translating the schedule into an oriented or directed graph, and working with the associated incidence matrix. A node in the graph considered represents a specific time at a specific terminal. The time chosen is the time of a local maximum in the "idle equipment inventory" at the terminal under consideration. There may be more than one "node" for each terminal; there will be at least one. An arriving run prior to a node time, which may be paired with a departing run after the node time, without violating minimality of equipment, is regarded as a link (arc) positively incident on that node. A departing run after a node time which can be paired with an arriving run prior to the node time is viewed as a link negatively incident on the node. This representation gives some idea of the versatility of graph theory in the study of

transportation problems.

With the incidence matrix constructed as described above, each column corresponds to one of the runs of the system, and each row corresponds to one of the nodes. One can then trace out a cycle, or a "pool", in the following manner. Connect a minus one with a plus one in the same row, and repeat, connecting plus ones and minus ones alternately within the same column and then the same row. When the first or starting entry is connected the second time, a cycle is formed. The pairing of runs in this manner is both feasible and maintains minimality of equipment.

BIBLIOGRAPHY

1. Koopmans, T. C., and Reiter, S., "A Model of Transportation" Activity Analysis of Production and Allocation, Cowles Commission Monograph 13, John Wiley & Sons, Inc., New York, 1951.

2. Robinson, Julia and Walsh, John, "Routing of Empties for Fixed-Schedule Transportation", The Rand Corporation, RM - 406, 12 June 1950.

3. Bartlett, T. E., "An Algorithm for the Minimum Number of Transport Units to Maintain a Fixed Schedule", Naval Research Logistics Quarterly, Volume 4, Number 2, June 1957.

4. Dantzig, G. B., and Fulkerson, D. R., "Minimizing the Number of Tankers to Meet a Fixed Schedule", Naval Research Logistics Quarterly, Volume 1, Number 3, September 1954.

5. Bartlett, T. E., and Charnes, A., Cyclic Scheduling and Combinational Topology: "Assignment and Routing of Motive Power to Meet Scheduling and Maintenance Requirements - Part II, Generalization and Analysis", Naval Research Logistics Quarterly, Volume 4, Number 3, September 1957.

6. Hunt, Ralph B. and Rosholdt, Earling F., "Determining the Size of an Ocean Shipping Pipeline Meeting a Time-Phased Military Requirement Schedule Under Convoy Conditions", The George Washington University Logistics Research Project, Serial T-113/60, 17 February 1960.

CHAPTER V

MISCELLANEOUS TECHNIQUES

Up to this point we have dealt with a rather restricted group of transportation problems and models. For the most part the models have been of the network type and descriptive of an entire transportation system. In this chapter we shall give examples of the application of other techniques to problems in transportation. Specifically we shall cite examples using queuing theory, the Monte-Carlo method, and engineering analysis.

Among the earliest models employed in the study of transportation systems was the queue, or waiting-line, model. Examples of queues in transportation systems might include automobiles at a stop sign or a toll station, freight cars in a freight yard, airliners circling an airport awaiting clearance to land, or ships in harbor awaiting unloading. In a different but related field this model has been applied to telephone calls at switchboards and messages in communications relay stations.

Waiting line theory is strongly related to probability theory, since both the arrival times of "customers" and the service time for an individual customer are stochastic variables. This is true to a large extent even though arrivals might be "scheduled". The usual practice is to study the so-called steady state; that is, one assumes the queue has been in operation long enough for transient effects to have diminished to a negligible point. One usually knows, or can soon estimate with acceptable accuracy, the mean arrival rate of customers and the mean service time. The specification of the distribution of these random variables is more difficult. However, the most widely used assumption is that the

number of arrivals within a specified interval has a Poisson distribution and that service times have a negative exponential distribution. With these assumptions and inputs, the model can be analyzed to predict mean holding time - waiting time plus service time -, mean queue length, and the probability a customer will have to wait.

A good example of the application of queuing theory to a transportation problem is contained in an article by Leslie C. Edie [1] of The Port of New York Authority, in which he discussed traffic delays at toll booths of several tunnels serving the city of New York.

In many problems involving queues, solutions in closed form are difficult, if not impossible, to obtain. Such problems are sometimes amenable to a certain amount of analysis by the Monte-Carlo method. In using this method, one develops a model of the system he wishes to study and then simulates arrivals from time to time. The arrival times are assigned at random in accordance with previously observed arrival statistics. With the model the investigator can determine the effect of varying certain parameters in the system. He can, for instance, introduce an additional service channel or allow priorities. With the simulation technique, the researcher can get data corresponding to months of actual operation quickly and cheaply.

Roger R. Crane, Frank B. Brown, and R. O. Blanchard [2] have reported on the application of this method to a railroad classification yard. These authors developed a model by the method described above and, using simplified statistical assumptions, isolated areas for more careful study. In the two areas reported they determined the effect of substituting parallel processing for series processing and of changing the "rules" regarding the

use of switch engines. Considerable savings were realized in both instances. "Processing" in this context consisted of inspection and classification.

For our last example we have chosen the Cargo-Handling Research Project in the Department of Engineering, University of California at Los Angeles. This research is sponsored by the Office of Naval Research and the Maritime Administration. Though by no means limited thereto, this project has made use of engineering analysis, statistical methods, and the Monte-Carlo technique. The project has been reported in detail in a series of numbered reports, (references [4], [5], [6], [7]). These are not generally available. R. R. O'Neill reported on the project and discussed the analysis and Monte-Carlo simulation of cargo handling before the Purdue University First Transportation Research Symposium, in February 1957. This presentation, along with others given at the symposium, was published in Naval Research Logistics Quarterly [3] in September 1957.

The transfer of cargo between ship and shore has long been recognized as the major bottleneck in the operations of the shipping industry. The basic objective of the research at the University of California at Los Angeles has been the effecting of improvements to the cargo-handling system.

For the analysis of cargo handling, the project faced the problem of choosing a level of description which would yield to analysis and yet lead to useful results. The level chosen considered the cargo-handling system as a series of transporting links in which each link was the sum of all the movements required of a carrier in transferring a load of

cargo from one point to another. In a land-to-water sequence the links studied were: (a) the shed link, (b) the wharf link, (c) the hook link, (d) the hold intermediate link, and (e) the hold link.

With the cargo-handling system idealized in the foregoing manner, certain hypotheses were made concerning the interrelations existing among the various links. Two examples of such hypotheses are:

> Hypothesis I: If there is a group of transporting links connected in series, then there is one link which has the least mean induced delay and is the slowest or controlling link.

> Hypothesis II: The relationship between the relevant factors which affect the mean activity time of a link can be formulated so as to make prediction possible.

Both of these hypotheses were used as the basis for a field study conducted in the summer of 1953, primarily in the ports of Los Angeles and San Francisco. Even though a random, work-sampling technique was used to obtain data, there were 41,218 individual observations recorded. This gives some idea of the magnitude of data-gathering problems.

It is noted that at the gross level a port terminal can be thought of as a node with a land carrier as the link on one side and a ship as the link on the other side. The approach used in cargo-handling study illustrates the meaning of a remark made earlier concerning the necessity of building models in a degree of detail consistent with the problem to be studied.

BIBLIOGRAPHY

1. Edie, Leslie C., "Traffic Delay at Toll Booths", Journal of the Operations Research Society of America, Volume 2, Number 2, May 1954.

2. Crane, Roger R., Brown, Frank B., and Blanchard, R. O., "An Analysis of a Railroad Classification Yard", Journal of the Operations Research Society of America, Volume 3, Number 3, August 1955.

3. O'Neill, R. R., "Analysis and Monte-Carlo Simulation of Cargo Handling", Naval Research Logistics Quarterly, Volume 4, Number 3, September 1957.

4. An Engineering Analysis of Cargo Handling, Report 53-21, Department of Engineering, University of California, Los Angeles, August 1953.

5. An Engineering Analysis of Cargo Handling - II, The Field Study of 1953, Report 55-2, Department of Engineering, University of California, Los Angeles, November 1954.

6. Davis, Harold, and Weinstock, Jack, An Engineering Analysis of Cargo Handling - III, Report 56-34, Department of Engineering, University of California, Los Angeles, July 1956.

7. O'Neill, Russell R., An Engineering Analysis of Cargo Handling - V, Simulation of Cargo Handling Systems, Report 56-37, Department of Engineering, University of California, Los Angeles, September 1956.

APPENDIX A

ALGORITHMS FOR THE HITCHCOCK DISTRIBUTION PROBLEM

1. The Dantzig Algorithm.

In this appendix, we shall describe the essential elements of Dantzig's algorithm for solving an m x n transportation problem based on his simplex method. See Chapter I. We shall then work out a simple problem using this method. We shall make similar presentations of the algorithms of Ford and Fulkerson, and Munkres, working out the same problem in each instance. The problem is 4 x 5 (four sources, five destinations) and the numbers are quite simple. We hoped to choose one which could be followed easily and yet was sufficiently difficult to illustrate the methods. The problem is stated below:

There are four sources with amounts to be shipped as follows:

$$a_1 = 10$$

$$a_2 = 60$$

$$a_3 = 40$$

$$a_4 = 90$$

There are five destinations with amounts required as follows:

$$b_1 = 20$$

$$b_2 = 40$$

$$b_3 = 70$$

$$b_4 = 30$$

$$b_5 = 40$$

We note $\sum a_i = \sum b_j = 200$.

The cost of transporting a unit amount of commodity from source i to destination j is given as the ij entry in the following matrix:

	1	2	3	4	5
1	4	1	6	5	1
2	2	3	5	3	2
3	5	2	3	4	4
4	4	1	4	1	2

Our problem is to find:

$X = [x_{ij}]$ such that

$x_{ij} \geq 0 \qquad i = 1,2,3,4 \quad j = 1,2,3,4,5 \qquad 1.1$

$\sum_j x_{ij} = a_i \qquad i = 1,2,3,4 \qquad 1.2$

$\sum_i x_{ij} = b_j \qquad j = 1,2,3,4,5 \qquad 1.3$

$\sum_i \sum_j c_{ij} x_{ij} = \text{Minimum}$

Before commencing the algorithm, we shall state, without proof, some useful theorems relating to the problem. For proofs the reader is referred to Dantzig [1] and Gass [2] .

Theorem 1. The transportation problem has a feasible solution.

Theorem 2. A solution of at most $m + n - 1$ positive x_{ij}'s exists.

Theorem 3. If the a_i and b_j are all non-negative integers, then every basic feasible solution has integral values.

Theorem 4. A finite feasible solution always exists.

For a more general case with m sources and n destinations, if we write out the equations 1.2 and 1.3 we have $m + n$ equations in mn unknowns. But,

since we have specified $\sum a_i = \sum b_j$, there is one redundant equation and the system may be reduced to $m + n - 1$ linearly indpendent equations. A basic feasible solution then has $m + n - 1$ positive x_{ij}'s. If a feasible solution is found with less than $m + n - 1$ positive x_{ij}'s, degeneracy is indicated. The difficulty can be averted by perturbing the problem in the following manner: add a small positive amount ϵ to each a_i, leave all b_j's unaltered except for b_n, and add $n\epsilon$ to b_n. This procedure need not be employed until degeneracy is indicated. Solve the problem for a minimum, basic feasible solution, using the ϵ's if necessary.

After the solution is obtained allow each ϵ to go to zero. The solution is still valid. In our problem, since each a_i and b_j is a positive integer, we are assured of a solution involving only positive integers.

To explain the computational procedure we shall work out our problem step by step. As the initial step, we write down the matrix of direct cost coefficients and a blank assignment matrix. We shall habitually write the column of a_i's to the right of the assignment matrix, and a row of b_j's below the matrix. This procedure will serve to keep the problem before us.

[illegible] Cost Matrix

4	1	6	5	1
2	3	[illegible]	[illegible]	2
5	[illegible]	5	[illegible]	4
4	1	4	1	

Assignment Matrix

					a_i
					10
					60
					40
					90
b_j	20	40	70	30	40

Working with the assignment matrix we start in cell (1,1) and assign the minimum $[a_1, b_1] = a_1 = 10$ units to this cell. We can [illegible] further [illegible] move to cell (2,1) and assign the minimum $(a_2, b_1 - 10) = 10$ units to this cell. We have now completed assignments in column one. We next move to cell (2,2) and make the assignment minimum $[40, 60 - 10] = 40$. Continuing in this way, we obtain the following basic, feasible solution.

	1	2	3	4	5
1	10				
2	10	40	10		
3			40		
4			20	30	40

In this example we have $m + n - 1 = 8$ positive x_{ij}'s. If after having made one of our assignments x_{ij}, the amount remaining in the column j and that remaining in the row i had vanished simultaneously, we would have had a degenerate solution, except for the case $i = m$ and $j = m$. In case of degeneracy, inclusion of the ϵ perturbation, as described above, will remove it; for then no partial sum of the a_i's and b_j's can equal any a_i or b_j.

For comparison purposes we note here the cost of our first basic feasible solution.

$$C_1 = 10(4) + 10(2) + 40(3) + 10(5) + 40(3) + 20(4) + 30(1) + 40(2) = 540$$

Next we compute, for the basic feasible solution just obtained, its indirect cost matrix. Each element $\bar{c}_{ij}$ of the indirect cost matrix will be the sum of an indirect cost u_i associated with the source (row) and an indirect cost v_j associated with the destination (column). Thus

$$\bar{c}_{ij} = u_i + v_j \qquad \text{for all } i \text{ and } j.$$

For those pairs i, j employed in the trial solution

$$\bar{c}_{ij} = c_{ij}$$

From this latter equation we can get some of the entries for the indirect - cost matrix. In our example we get the incomplete matrix below.

Indirect - Cost Matrix (incomplete)

4				
2	3	5		
		3		
		4	1	2

we can now let $u_1 = 4$ and $v_1 = 0$, arbitrarily, and compute the remaining u_i's and v_j's from the relation

$$\bar{c}_{ij} = u_i + v_j \qquad \text{for } \underline{\text{all}}\ i \text{ and } j.$$

Doing so we obtain the indirect cost matrix below. Note $v_1 = 0$ hence by the equation $u_2 + v_1 = \bar{c}_{21} = 2$, u_2 must equal 2. Then $v_2 = 1$ and so on, until all u_i and v_j are determined. The remaining $\bar{c}_{ij}$ are computed by using the relation, $\bar{c}_{ij} = u_i + v_j$.

Indirect Cost Matrix

						u_i
	4	5	7	4	5	4
	2	3	5	2	3	2
	0	1	3	0	1	0
	1	2	4	1	2	1
v_j	0	1	3	0	1	

In order to facilitate comparison of corresponding entries in the direct and indirect cost matrices, we shall write them together as one matrix, called the <u>cost matrix</u>. In each cell of the cost matrix we shall write $\bar{c}_{ij}$ in the upper half and c_{ij} in the lower half. In our sample problem this results in the following cost matrix for step 1.

Cost Matrix
(Step 1)

						u_i
	4 \ 4	5 \ 1	7 \ 6	4 \ 5	5 \ 1	4
	2 \ 2	3 \ 3	5 \ 5	2 \ 3	3 \ 2	2
	0 \ 5	1 \ 2	3 \ 3	0 \ 4	1 \ 4	0
	1 \ 4	2 \ 1	4 \ 4	1 \ 1	2 \ 2	1
v_j	0	1	3	0	1	

Entries above the diagonal are $\bar{c}_{ij}$ $\quad i = 1,\ldots,4$ $\quad j = 1,\ldots,5$

Entries below the diagonal are c_{ij} $\quad i = 1,\ldots,4$ $\quad j = 1,\ldots,5$

The additional column to the right is the column of u_i's.

The additional row below is the row of v_j's. These additions are included for convenience.

We now look at those cells in which $\bar{c}_{ij} > c_{ij}$. If there are none the basic feasible solution we have chosen is a minimum cost solution and our problem is solved. If $\bar{c}_{ij} > c_{ij}$ for some i and j, we wish to determine

$$M = \max_{ij} \bar{c}_{ij} - c_{ij}$$

In our case $M = 4$, for cells (1,2) and (1,4), encircled in our cost matrix, step 1. We wish now to assign to one cell corresponding to M, an amount Θ_1[1] as large as possible consistent with the requirement that all x_{ij} be non-negative, and that $\sum_j x_{ij} = a_i$ and $\sum_i x_{ij} = b_j$. If we increase the assignment in cell (i_o, j_o) by Θ_1, we must decrease by a corresponding amount the assignment in some other cell of row i_o, in order to meet the latter constraints above. This adjustment will cause an unbalance in a different column; so we must continue this compensating procedure until we complete a cycle by decreasing by Θ_1 the assignment in some cell (i, j_o).

In our example we have two candidates for the assignment Θ_1, cells (1,2) and (1,4). We arbitrarily choose cell (1,2) and formally enter Θ_1 therein, because we do not yet know its value. Making the required adjustments we have the assignment matrix in the following interim form. The cycle we have followed, alternately adding and subtracting Θ_1, is (1,2), (1,1), (2,1), (2,2).

1. The subscript 1 refers to the first iteration. In subsequent iterations we use Θ_2, Θ_3..., until the problem is solved.

						a_i
	$10-\Theta_1$	Θ_1				10
	$10+\Theta_1$	$40-\Theta_1$	10			60
			40			40
			20	30	40	90
b_j	20	40	70	30	40	

In each iteration, at this point there will always be at least two entries, $[x_{ij} - \Theta]$. We make Θ as large as possible, keeping all $x_{ij} \geq 0$. In the present example $\Theta_1 \leq 10$ in order for x_{11} to remain non-negative. Hence we set $\Theta_1 = 10$ and get the new assignment matrix shown below.

Assignment Matrix (step 2)

						a_i
		10				10
	[illegible]	30	10			60
			40			40
			20	30	40	90
b_j	20	40	70	30	40	

This procedure has resulted in a new, basic feasible solution. We have made x_{12} positive and $x_{11} = 0$. If, at any stage, the introduction of Θ units to one route causes two or more positive x_{ij} to vanish, degeneracy is indicated. The inclusion of the ϵ perturbation, previously described, will eliminate the degeneracy.

Below we rewrite our assignment matrix (step 2) and its associated cost matrix.

Step 2

Cost Matrix

						u_i
	0, 4	1, 1	3, 6	0, 5	1, 1	1
	2, 2	[illegible]	5, 5	2, 3	3, 2	3
	0, 5	1, 2	3, 3	0, 4	1, 4	1
	1, 4	2, 1	4, 4	1, 1	2, 2	2
v_j	-1	0	2	-1	0	

Assignment Matrix

						a_i
		10				10
	20	$30-\Theta_2$	$10+\Theta_2$			60
			40			40
		Θ_2	$20-\Theta_2$	30	40	90
b_j	20	40	70	30	40	

We have again encircled two possible routes for the introduction of Θ_2. Again our choice is somewhat arbitrary, but route (4,2) has the lesser direct cost; hence, we choose it. We have indicated the plus and minus Θ_2's above. From the assignment matrix we see $\Theta_2 \leq 20$, in order that x_{43} not be non-negative. We let $\Theta_2 = 20$. Continuing this process, we go through the following iterations.

Step 3

						u_i
	0, 4	1, 1	3, 6	1, 5	2, 1	1
	2, 2	3, 3	5, 5	3, 3	4, 2	3
	0, 5	1, 2	3, 3	1, 4	2, 4	1
	0, 4	1, 1	3, 4	1, 1	2, 2	1
v_j	-1	0	2	0	1	

						a_i
		10				10
	20	$10-\Theta_3$	30		Θ_3	60
			40			40
		$20+\Theta_3$		30	$40-\Theta_3$	90
b_j	20	40	70	30	40	

$\Theta_3 = 10$

Step 4

						u_i
	2 / 4	1 / 1	5 / 6	1 / 5	2 / 1	1
	2 / 2	1 / 3	5 / 5	1 / 3	2 / 2	1
	0 / 5	-1 / 2	3 / 3	-1 / 4	0 / 4	-1
	2 / 4	1 / 1	5 / 4	1 / 1	2 / 2	1
v_j	1	0	4	0	1	

						a_i
		$10-\Theta_4$			Θ_4	10
	20		30		10	60
			40			40
		$30+\Theta_4$		30	$30-\Theta_4$	90
b_j	20	40	70	30	40	

$\Theta_4 = 10$

Step 5

						u_i
	1 / 4	0 / 1	4 / 6	0 / 5	1 / 1	1
	2 / 2	1 / 3	5 / 5	1 / 3	2 / 2	2
	0 / 5	-1 / 2	3 / 3	-1 / 4	0 / 4	0
	2 / 4	1 / 1	5 / 4	1 / 1	2 / 2	2
v_j	0	-1	3	-1	0	

						a_i
					10	10
	20		$30-\Theta_5$		$10+\Theta_5$	60
			40			40
		40	Θ_5	30	$20-\Theta_5$	90
b_j	20	40	70	30	40	

$\Theta_5 = 20$

Step 6

						u_i
	1 / 4	1 / 1	4 / 6	1 / 5	1 / 4	1
	2 / 2	2 / 3	5 / 5	2 / 3	2 / 2	2
	0 / 5	0 / 2	3 / 3	0 / 4	0 / 4	0
	1 / 4	1 / 1	4 / 4	1 / 1	1 / 2	1
v_j	0	0	3	0	0	

						u_i
					10	10
	20		10		30	60
			40			40
		40	20	30		90
b_j	20	40	70	30	40	

For step 6, $M = \max_{ij} (\bar{c}_{ij} - c_{ij}) = 0$, and the assignment matrix represents a solution. The cost is

$$C = 10(1) + 20(2) + 10(5) + 30(2) + 40(3) + 40(1) + 20(4) + 30(1)$$

$$= 10 + 40 + 50 + 60 + 120 + 40 + 80 + 30$$

$$= 430$$

2. The Ford - Fulkerson Algorithm.

Two matrices are carried along:

(1) The cost-weight matrix. This is the original cost matrix C with "weights" attached to the rows and columns.

(2) An additional matrix of the same size as C, called the zero matrix. This matrix is used in making the assignments, i.e. choosing the positive x_{ij}'s. With each row we associate a "surplus" and with each column a "shortage"; initially these are the a_i and b_j of the original problem. Consider the problem previously given which is summarized in the following tableau.

	4	1	6	5	1
	2	3	5	3	2
	5	2	3	4	4
	4	1	4	1	2

a_i \ b_j	20	40	70	30	40
10					
60					
40					
90					

We obtain the initial cost-weight matrix and the initial zero matrix in the following manner. Prepare two matrices of the same size. One is the original cost matrix $C = [c_{ij}]$. Leave the other blank

except to the left list initial surpluses a_i and across the top list initial shortages b_j.

-1	4	1	6	5	1
-2	2	5	5	3	2
-2	–	2	3	4	4
-1	4	1	4	1	2

	20	40	70	30	[illegible]
10					
60					
[illegible]					
90					

Scan each row of the cost matrix, pick out the least entry and enter its negative as the weight for that row. (This step has been done above.) Next place circles in the zero matrix corresponding to each appearance of these least values in the cost matrix. (This step has also been done above).

Then with each column in the zero matrix that contains a circle associate a weight of zero and enter these weights in appropriate positions in a row above the cost matrix. For each column not possessing a circle (column three in the example, see matrix below) compute each entry plus its row weight. Take the minimum of these results and enter its negative as the column weight for the column. Enter circles in the zero matrix in the cells corresponding to these minimal values. With a weight attached to each row and each column, our cost matrix is now better described as a cost-weight matrix. It will hereafter be so designated. Results of the foregoing step are shown below for our example.

	0	0	-1	0	0
-1	4	1	6_{5}	5	1
-2	2	3	5_{3}	3	2
-2	5	2	3_{1}	4	4
-1	4	1	4_{3}	1	[illegible]

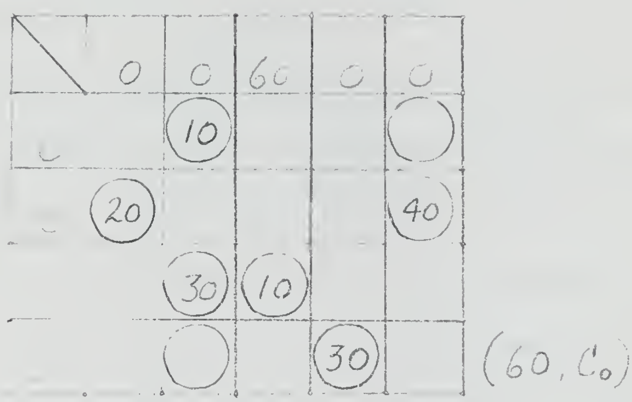
	20	40	70	30	4[illegible]
10					
60					
40					
90					

The next step may be thought of as partially satisfying shortages in the columns from surpluses in the rows, using only circled positions as possible shipping routes. This step is accomplished as follows: Scan the first row for the first circle appearing in it. Determine the minimum of the surplus for that row and the shortage for that column. Enter this number in the circle and simultaneously reduce the shortage and surplus by that amount. Then continue to the second circle and so on. When the first row is completed go to the second. Repeat until all rows have been scanned. Applying this procedure to the zero matrix of our problem yields:

	0	0	60	0	0
0		10			
0	20				40
		30	1[illegible]		
60				30	

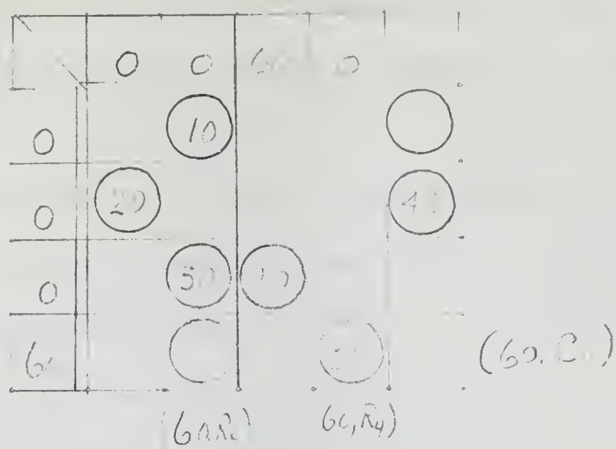
Now we commence an iterative procedure which will eventually produce a solution. This procedure starts with a labeling process which results in either a breakthrough or non-breakthrough. The procedure for each alternative will be discussed below but first let us describe the labeling process.

Start by labeling the rows which still have surpluses. (In our example there is only one.) Use the label (ℓ, C_0) for each such row, where ℓ is the amount of the surplus in that row. C_0 refers to the column of surpluses. Applied to our example this step gives the following diagram



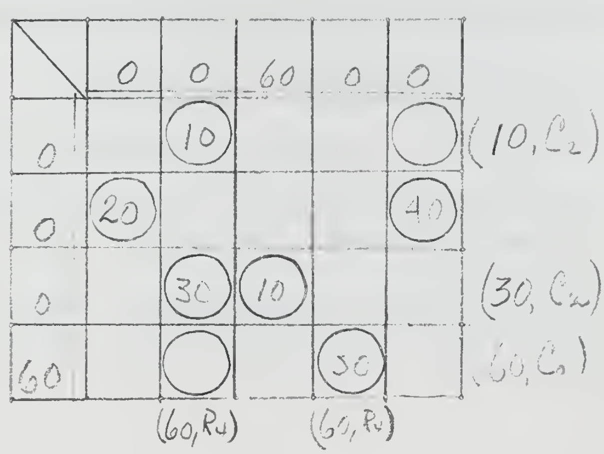
Hereafter the number ℓ will be referred to as the "potential flow" associated with the row.

Next we turn to columns; initially, none of them are labeled at this point. We take each labeled row and sweep along it looking for circles in currently unlabeled columns. If such a circle is found label the column containing it (f, R_i) where f is the potential flow associated with the row being scanned and R_i identifies the row. Results of this step are shown below.



The number f will hereafter be called the "potential flow" associated with the column.

When the rows have all been scanned, turn to the newly labeled columns. Scan each labeled column for circles containing positive entries but not appearing in a previously labeled row. Any row in which such a circle lies is given a label (ℓ, C_j) where C_j is the column being scanned and ℓ is the smaller of the potential flow of the column and the entry in the circle. Column scanning differs from row scanning in two ways. The circles must have positive entries, and ℓ is computed differently. Applying this step to our example, yields the following diagram:



Having scanned all newly labeled columns we turn again to the rows scanning the newly labeled ones as before. Continue this procedure scanning newly labeled rows and columns until either a breakthrough is achieved or no new labeling is possible. When scanning row #3 we achieved a breakthrough (i.e. we labeled a column that had a shortage). At that point the matrix was labeled as below.

	0	0	60	0	0	
0		10				$(10, C_2)$
0	20				40	
0		30	10			$(30, C_2)$
1				30		$([illegible], C_1)$
		$(60, R_{[illegible]})$	$(30, R_3)$	$(60, R_4)$	$(10, R_1)$	

Breakthrough

We stop labeling when a breakthrough is achieved and make adjustments in the allocations. Suppose we have labeled a column which has a shortage, s, with the label (f, R_i). In our case $s = 60$ and the label is $(30, R_3)$. This means that we can increase the flow to the labeled column by an amount $h = \min [s, f]$. The following procedure is used.

(1) Decrease the shortage by an mount h. (30 in our case)

(2) Look at the label (f, R_i) and increase the entry in the circle where row i and the labeled column intersect by h units.

(3) The label on row i designates a certain column, j. Decrease the entry in the circle where row i meets column j by h units.

(4) Obtain a new row from the label on column j and increase, continue alternately increasing and decreasing the amounts in circles by h units until a row is reached whose label is (ℓ_0, C_0).

(5) When the indicated surplus is reduced by h units, this process is completed.

Applied to our example these steps produce the following matrix.

Arrows indicate the "path" we followed in the adjustment process.

	0	0	60-30 =30	0	0
0		10			
0	20				40
0		30-30 =0	30+10 =40		
60-30 =30		0+30 =30			

If, at this point, all shortages and surpluses are zero, the problem is solved. Otherwise, remove all previously applied labels and commence the labeling process anew.

Next time around, we exhaust all possible labelings without achieving further breakthroughs and have the matrix shown below.

	0	0	30	0	0	
0		10				(10, C_2)
0	20				40	(10, C_4)
0			40			
30		30		30		(30, [illegible])
	(10, R_2)	(30, R_4)		(30, R_4)	(10, R_1)	

Although we have labeled all except one row and one column, we have not achieved breakthrough. At this point we turn our attention to the cost-weight matrix which provides a means of proceeding.

<u>Non-breakthrough</u>

We now change the weights on the cost-weight matrix by subtracting k units from the weights of the labeled rows and adding k units to the weights of the labeled columns. The value of k is determined by taking it as large as possible subject to the constraint $c_{ij} + w_i + w_j \geq 0$ where c_{ij} is the ij entry in the cost matrix and w_i is the row weight, w_j is the column weight. The determination of k can be accomplished as follows:

Take up the cost-weight matrix but ignore for the time being all labeled columns. Shading, covering, or some other distinguishing marking is recommended for the labeled columns. For each entry in each labeled row, compute the algebraic sum of it, its row weight, and its column weight. The minimum value obtained this way is k. Enter circles in the

zero matrix in positions corresponding to this value. We apply this step to our example. Check marks denote labeled rows/columns.

Cost-Weight Matrix

	0	0	-1	0	0	✓
-1	4	1	6 $_5$	5	1	✓
-2	2	3	5 $_2$	3	2	
-2	5	2	3	4	4	
-1	4	1	4 $_2$	1	2	✓

Zero Matrix

	0	0	30	0	0
0		10			
0	20		◇		40
0			40		
30		30	◇	30	

k = 2

In our case all columns have been labeled except 3. We need consider only positions (1,3), (2,3), and (4,3). New circles are placed in positions (2,3) and (4,3), indicated temporarily by diamonds in the zero matrix. Subtract k from each labeled row's weight and add k to each labeled column's weight. Now looking at the zero matrix; we ignore all labeled rows and remove all circles remaining in labeled columns. Only the circle in position (3,2) is to removed in our case. It has been crossed out above. As a check on the procedure, this process should never lead to the removal of a circle with a positive entry.

The above procedures provide us a new zero matrix with which we recommence the labeling process. A new cost-weight matrix is also shown

	2	2	-1	2	2
-3	4	1	6_4	5	1
-4	2	3	5_2	3	2
-2	5	2	3	4	4
-3	4	1	4_2	1	2

	0	0	30	0	0
0		10			
0	20				40
0			40		
30		30		30	

With very little labeling we achieve breakthrough into column 3.

	0	0	30	0	0	
0		10				
0	20				40	
0			40			
30		30		30		$(30, C_0)$
		$(30, R_4)$	$(30, R_4)$	$(30, R_4)$		

This breakthrough provides us the needed 30 units flow to column 3, and the problem is solved. All shortages and surpluses are now zero.

The cost of this minimax solution is given as follows

$$C = [illegible] + [illegible] + (40)(2) + (40)(3) + (30)(1)$$

$$+ [illegible] + (30)(1)$$

$$C = 430$$

This is the same cost as obtained by the other methods, as it must be.

The above description is really more involved than the actual use of the algorithm. Once the ground rules are understood, the manipulation of the matrices is not difficult. The authors of this algorithm state they completed a 20 x 20 optimal assignment problem in about 30 minutes of hand computation. The same problem by the simplex method, required well over an hour.

3. The Munkres Algorithm.

Because of its basic similarity to the Ford - Fulkerson method, this algorithm will be stated completely, and then the example will be worked out. Munkres statement of the problem is given to speficy his notation.

Let:

$D = [d_{ij}]$ be an m x n matrix of non-negative integers,

r_i $(i = 1,2,\ldots,m)$ and c_j $(j = 1,2,\ldots,n)$ be positive integers such that $\sum_i r_i = \sum_j c_j = N$.

Determine values of x_{ij} which minimize the sum $\sum_{ij} x_{ij} d_{ij}$ subject to

$$x_{ij} \geq 0$$

$$\sum_i x_{ij} = c_j$$

$$\sum_j x_{ij} = r_i$$

The following statement of the algorithm is taken directly from Munkres' paper "Algorithms for the Assignment and Transportation Problems."

> We work with the cost matrix $D = [d_{ij}]$. In the course of the problem, we will distinguish certain lines of the matrix, calling them <u>covered</u> lines, and we will distinguish certain zero elements of the matrix by means of asterisks and primes. In addition, we will assign to each element of the matrix a non-negative quota x_{ij} which may be changed in the course of the problem. Each element of the matrix whose quota is positive will be called <u>essential</u> (these elements will always be zeros). At any stage of the problem, the number $c_j - \sum_i x_{ij}$ will be called the discrepancy of the jth column at that stage, and the number $r_i - \sum_j x_{ij}$ is the discrepancy of the ith row. These discrepancies will always be non-negative; when they all vanish, the corresponding quotas are a solution to the problem.

We shall construct a diagonal in each cell of the matrix and place the d_{ij} below and the corresponding x_{ij} above the diagonal. Initial discrepancies (the r_i and c_j) will be written to the left and above the matrix. Discrepancies at succeeding stages will be shown within dotted lines with row discrepancies to the right and column discrepancies below the cost matrix. We will denote row discrepancies as Δ_i, column discrepancies, as Δ_j, $i = 1, \ldots, m$, $j = 1, \ldots, n$. To avoid unnecessary clutter zero quotas will not be shown explicitly, but will be implied by a blank above the diagonal in each cell where a zero quota is assigned.

<u>Preliminaries</u>

All quotas are zero, no lines are covered, no zeros are starred or primed.

Subtract from each row of the matrix D its smallest entry, then subtract from each column of the resulting matrix its smallest entry.

Find a zero Z in the matrix. If the discrepancies of both its row and its column are positive, increase the quota assigned to Z until the smaller of the two discrepancies is reduced to zero. Repeat

for each zero in the matrix. Then cover every column whose discrepancy is zero.

Step 1. Choose a non-covered zero and prime it. Consider the row containing it. If the discrepancy of this row is positive go at once to step 2. Otherwise (if the row discrepancies is zero) cover the row. Then star each twice covered essential zero Z in the row and uncover Z's column. Repeat until all zeros are covered. Go to step 3.

Step 2. There is a sequence of alternately starred and primed zeros constructed as follows: Let Z_o denote the uncovered $0'$ (there is only one). Let Z_1 denote the 0^* in Z_o's column (if any). Let Z_2 denote the $0'$ in Z_1's row. Let Z_3 denote the 0^* in Z_2's column (if any). Similarly continue until the sequence stops at a $0'$, Z_{2k}, which has no 0^* in its column. Since no column contains more than one 0^* and no row more than one $0'$, this sequence is unique. It may contain only one element. The discrepancies of Z_o's row is positive, that of Z_{2k}'s column is positive, and the quota assigned each 0^* of the sequence $Z_o, \ldots, Z_{2k}$ is positive. Let h be the smallest of these positive members. Increase the quota of each $0'$ in the sequence by h, and decrease the quota of each 0^* in the sequence, by h. Erase all asterisks and primes. Uncover all rows and cover every column whose discrepancy is zero. Return to step 1.

Step 3. Let k denote the smallest non-covered element of the matrix; it will be positive. Add k to every covered row and subtract k from every uncovered column. Return to Step 1, without altering any asterisks, primes, or covered lines.

Our sample problem will be stated in Munkres' notation and solved according to his algorithm

$c_1 = 20$ $\qquad r_1 = 10$

$c_2 = 40$ $\qquad r_2 = 60$

$c_3 = 70$ $\qquad r_3 = 40$

$c_4 = 30$ $\qquad r_4 = 30$ $\qquad D_{ij} = \begin{array}{ccccc} 4 & 1 & 6 & 5 & 1 \\ 2 & 3 & 5 & 3 & 2 \\ 5 & 2 & 3 & 4 & 4 \\ 1 & 1 & 4 & 1 & 2 \end{array}$

$c_5 = 40$

Results of "preliminaries" ✓ indicates covered lines.

r_i \ c_j	20	40	70	30	40	Δ_i
10	3	10 0	4	4	0	0
[illegible]	20 0	1	2	1	40 0	0
[illegible]	3	30 0	10 0	2	2	0
[illegible]	3	0	2	30 0	1	60
Δ_j	0	0	60	0	0	
	✓	✓		✓	✓	

Result of Step 1.

$r_i \backslash c_j$	20	40	70	30	40	Δ_i
10	3	10 0	4	4	0	0
60	20 0	1	2	1	40 0	0
40	3	30 0^*	10 0'	2	2	0
90	3	0'	2	30 0	1	60
Δ_j	0	0	60	0	0	

Step 2. [illegible] is the [illegible] $(3,2)$, z_1 [illegible] $(3,3)$, z_{2k} $(3,3)$

$\delta = \min$ [illegible] $= 30$

Results of Step 2 and subsequent Step 1

$r_i \backslash c_j$	20	40	70	30	40	Δ_i
10	3	10 0	4	4	0	0
60	20 0	1	2	1	40 0	0
40	3	0	40 0	2	2	0
90	3	30 0	2	30 0	1	30
Δ_j	0	0	30	0	0	

Our second application of step one resulting only in priming the zero in [illegible] and [illegible] row [illegible]. [illegible] columns hence [illegible] covered. We proceed to step 3.

For application of step 3, our $k = 2$. Applying step 3 changes the entries in the cost matrix only in cells (1,3), (2,3), and (4,3). These become 2, 0, 0 respectively. We prime the zero in cell [illegible], apply step 2 with $h = 30$ and our solution is.

	10			
20				40
		40		
	30	30	30	

The cost is 430, as before.

The similarities between the latter two algorithms is apparent in the manner in which the solutions developed

BIBLIOGRAPHY

1. Dantzig, G. B., "Application of the Simplex Method to a Transportation Problem", Activity Analysis of Production and Allocation, John Wiley & Sons, Inc., New York, 1951

2. Gass, Saul I., Linear Programming Methods and Applications, McGraw-Hill Book Co., Inc., New York, 1958.

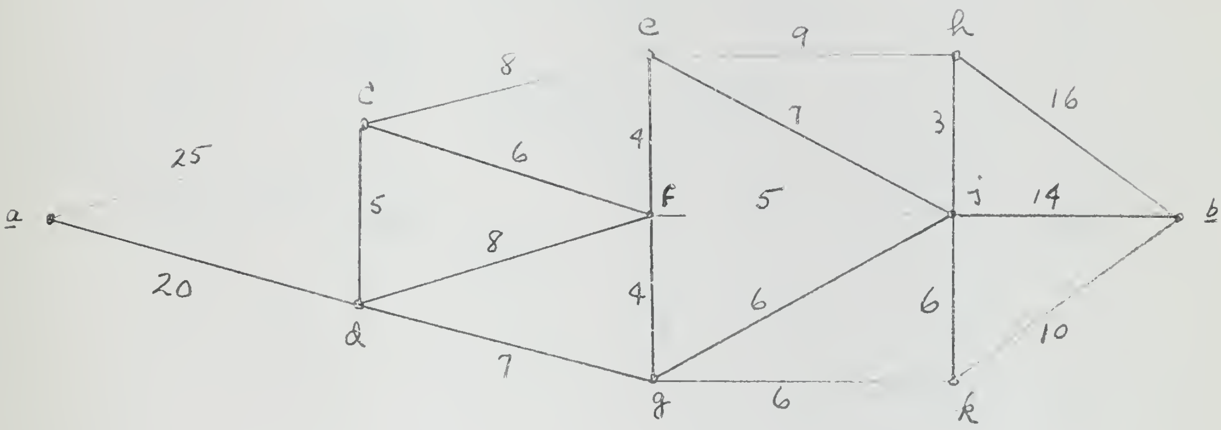
APPENDIX B

SOLUTIONS OF A MAXIMAL FLOW PROBLEM

1. General.

In this appendix we shall present a small capacitated network, and determine the maximal flow therein by using the techniques described in Chapter III. We shall work the problem first by the flooding technique and second by using the minimal-cut procedure. In each instance, we shall restate the essential features of the technique employed.

2. The Problem.



For the network N shown above determine the maximum steady-state flow, from origin a to terminal b.

3. The Boldyreff Flooding Technique.

Starting at the origin, assign sufficient traffic flows to all arcs leaving the origin to saturate them. View this set of junction points as new origins and, starting with the one subject to the greatest capacity constraint, schedule units whenever possible in the following order:

1. "Forward" - to new junction points through the outgoing arcs.
2. "Laterally" - to other points of the set.

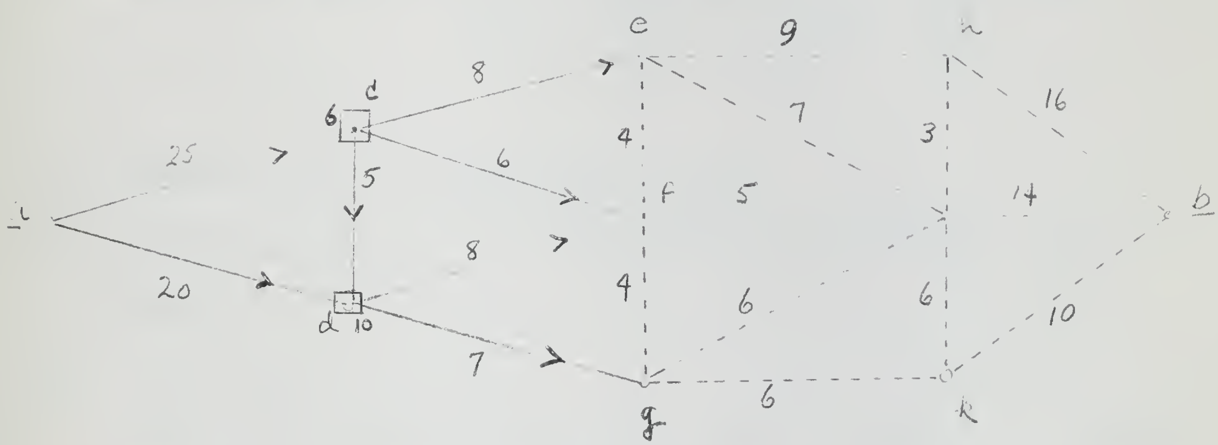
3 'Bottlenecked' - if units are left over after steps 1 and 2, i.e. if all outgoing and lateral arcs are saturated

Continue the above procedure until the accumulated flow covers the complete network and reaches the terminal

Eliminate bottlenecks by returning all excess units to the origin

The validity of the solution can be checked by inspection. If a maximal flow has been achieved, there will be no continuous unsaturated path extending from the origin to the terminal.

4 Solution by Polydyreff's Flooding Technique

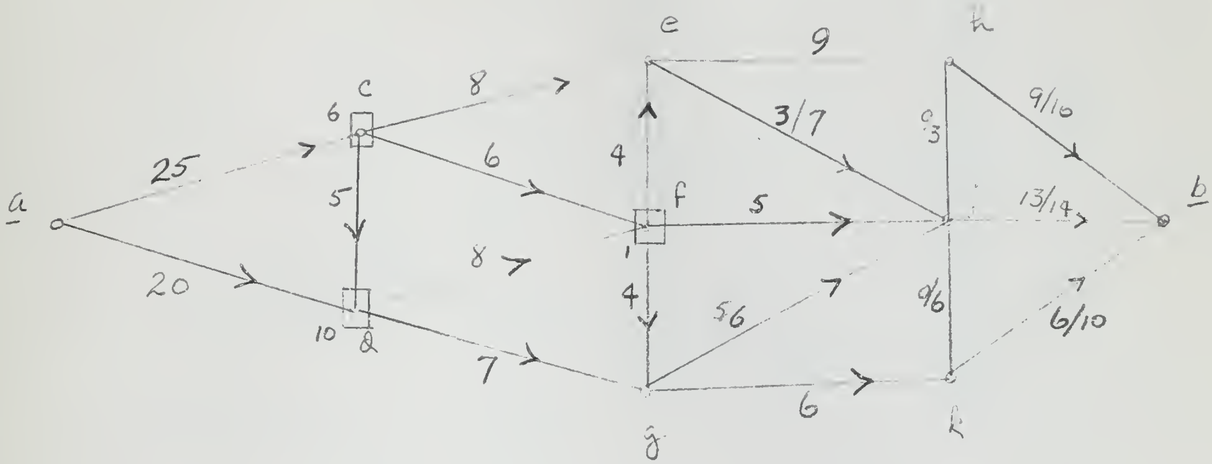


The above represents the results of the first two stages of flooding.

Bottlenecks: 6 at junction c, 10 at junction d.

All arcs arriving at points two arcs removed from the origin are saturated.

Further application yields the diagram below. Flow through unsaturated arcs is shown as the fraction of capacity actually used.



One additional unit is bottlenecked at point f.

Return bottlenecked units to origin, and the maximum flow is seen to be 28 units.

5. The Minimal-Cut Procedure.

Let T be the chain joining a and b which is topmost in N. Impose as large a flow as possible (T, k_1) on this chain, thereby saturating one of its arcs.

Subtract k_1 from the capacity of each arc of T.

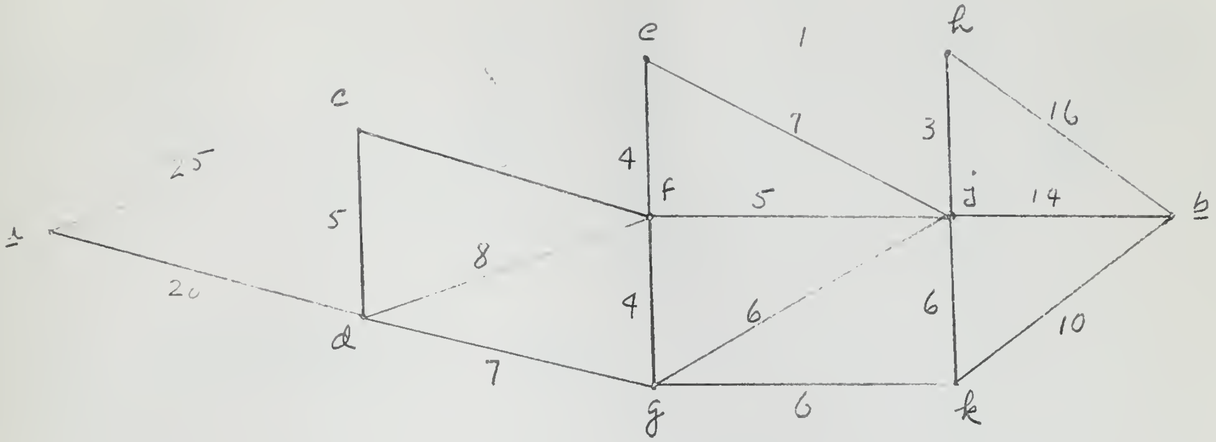
Remove the previously saturated arc whose capacity is now zero. Record k_1. Continue this procedure. Eventually, the graph will disconnect and the maximal flow is established as:

$$F = \sum k_i \qquad i = 1, \ldots, n$$

where: k_i is the amount of flow necessary to saturate the topmost chain in the i^{th} step, and n is the number of steps required for the graph to disconnect.

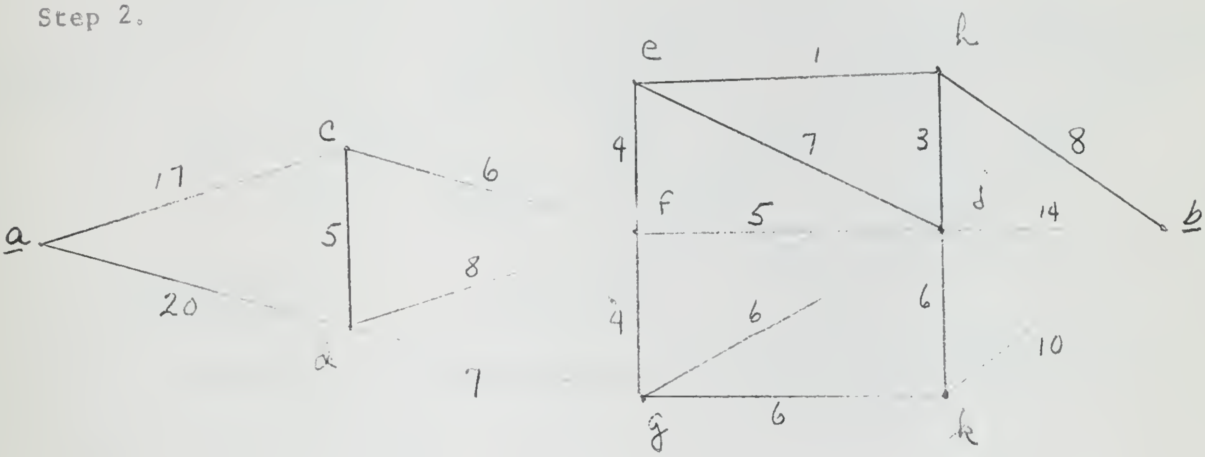
6 Solution by the Minimum-Cut Procedure

Step 1



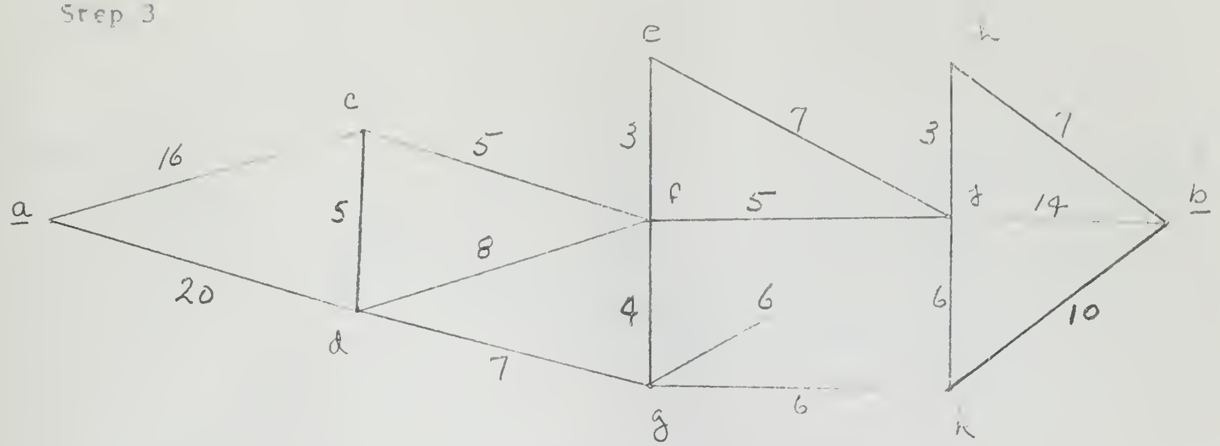
T_1 = (a, c, e, h, b,) k_1 = 8

Step 2.



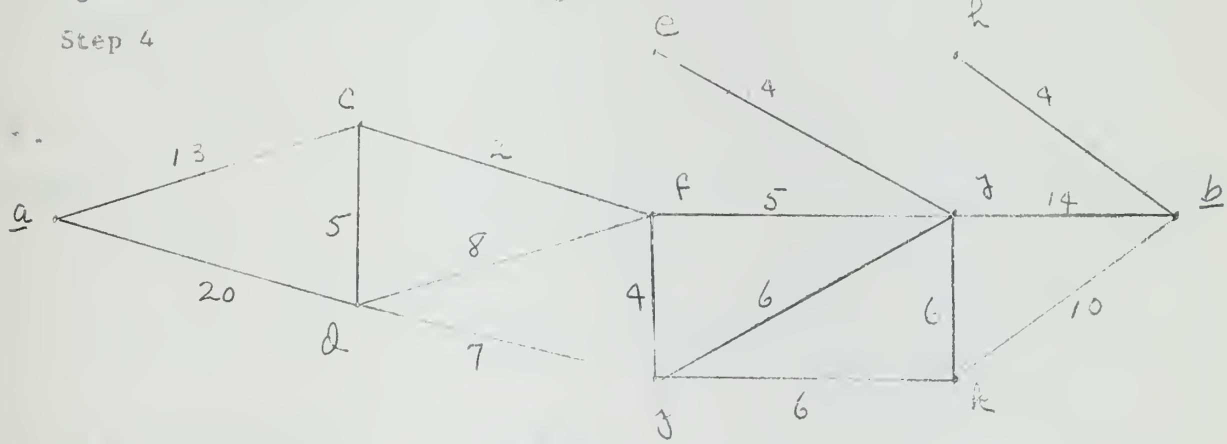
T_2 = (a, c, f, e, h, b) k_2 = 1

Step 3



$T_3 = (a, c, f, e, j, h, b)$ $\qquad k_3 = 3$

Step 4

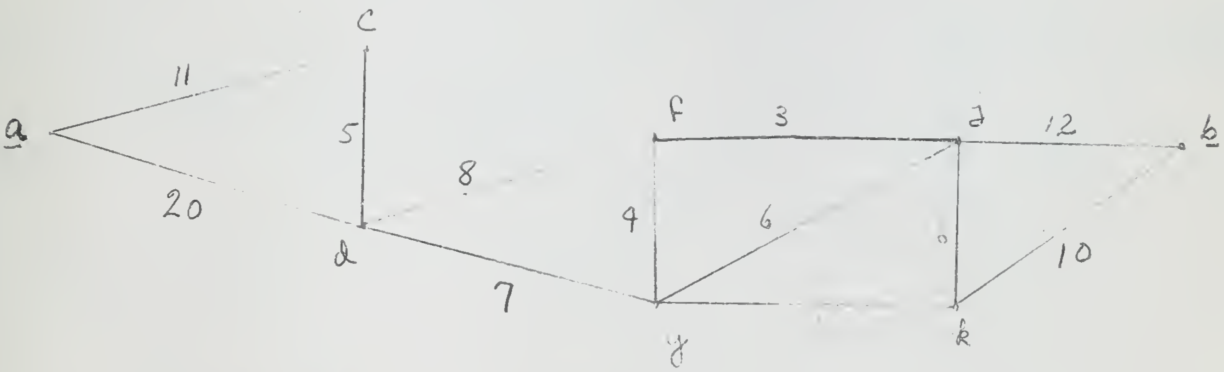


$T_4 = (a, c, f, j, b)$ $\qquad k_4 = 2$

Arcs ej and hb can no longer contribute to any chain flow from a to b

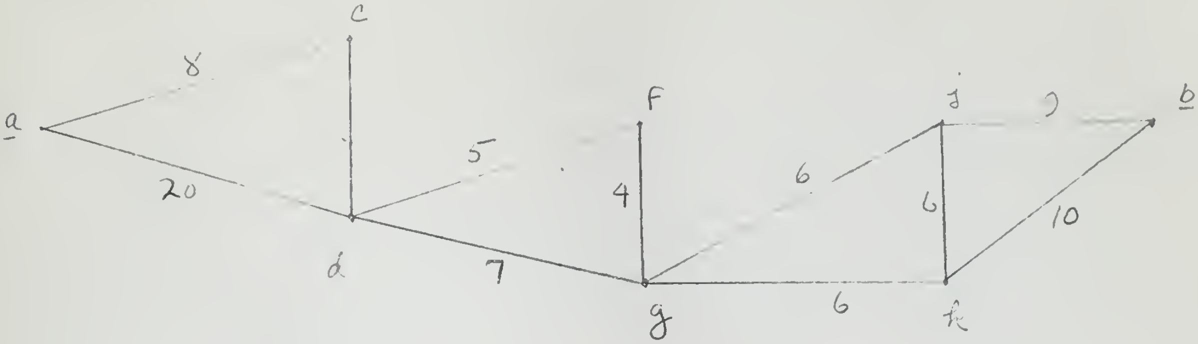
They will be eliminated.

Step 5



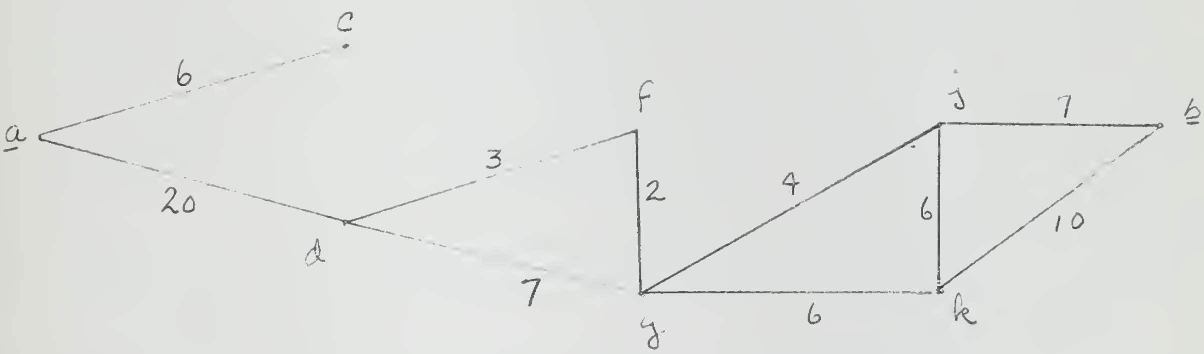
$T_5 = (a, c, d, f, j, b)$ $\qquad k_5 = 3$

Step 6



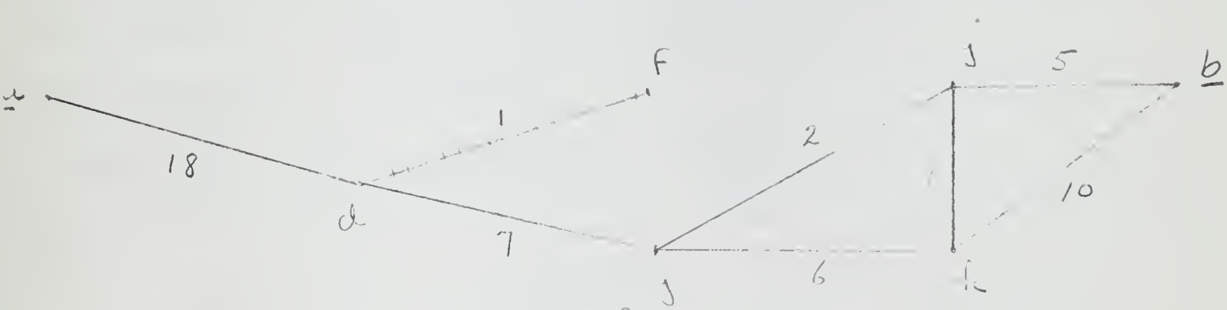
$T_6 = (a, c, d, f, g, j, b)$ $\qquad$ $k_6 = 2$

Step 7



$T_7 = (a, d, f, g, j, b)$ $\qquad$ $k_7 = 2$

Step 8



$T_8 = (a, d, g, j, b)$ $\qquad$ $k_8 = 2$

Step 9

$T_9 = (a, d, g, k, j, b)$ $k_9 = 3$

Step 10.

$T_{10} = (a, d, g, k, b)$ $k_{10} = 2$

On completion of this step the graph disconnects.

The capacity of N is given as.

$Cap(N) = \sum k_i = 8+1+3+2+3+2+2+2+3+2 = 28$ units.

APPENDIX C

GRAPH THEORY

In Chapters II, III, and IV of this thesis we have used the language of the mathematical theory of graphs. We include here definitions and theorems from this theory, in order to amplify and clarify some of our statements in the text.

1. DEFINITIONS.

A graph G is a finite, one-dimensional complex, consisting of vertices a, b, c, ..., e and arcs $\alpha(ab)$, $\beta(ac)$, ..., $\delta(ce)$.

An arc $\alpha(ab)$ joins it end vertices a and b; it passes through no other vertices and intersects no other arcs except in vertices.

A chain (path) is a set of distinct arcs of G which can be arranged as $\alpha(ab)$, $\beta(bc)$, ..., $\delta(gh)$ where the vertices a, b, c, ..., h are distinct. A chain does not intersect itself and it joins its end vertices, a and h.

A cycle is a set of distinct arcs that can be ordered as $\alpha(ab)$, $\beta(bc)$, ..., $\gamma(ef)$, $\delta(fa)$, the vertices being distinct as in the case of a chain.

A graph is connected if each pair of vertices is joined by a chain.

A forest is a graph containing no cycles, and a tree is a connected forest.

We may associate with each arc of a graph a positive number called its capacity.

The graph G together with the capacities of its individual arcs, is called a network.

An oriented or directed graph is a graph in which each arc has a direction specified. In such a graph, $\alpha(ab)$ and $\alpha'(ba)$, if they both exist, are distinct.

In some applications with oriented graphs it is desirable to distinguish certain subsets of vertices as origins and certain others as destinations. Other names used in this connection are origins and terminals and sources and sinks. The word terminals is also often used to distinguish a subset of nodes in a network, the remaining nodes being junction points.

A two-terminal network is a network in which two points are rooted and distinguished both from each other and from the remaining points. Usually the first of these points is called the <u>source</u> and the second is called the <u>sink</u>.

A <u>chain flow</u> from a source <u>a</u> to sink <u>b</u> is a couple (C, k) composed of a chain joining <u>a</u> and <u>b</u> and a non-negative number <u>k</u> representing the flow along C from <u>a</u> to <u>b</u>.

A <u>network-flow</u> is a collection of chain flows which has the property that the sum of the numbers <u>k</u> of all chain flows that contain any arc is no greater than the capacity of that arc. If equality holds the arc is said to be <u>saturated</u> by the flow.

The flow in a two-terminal network is said to be <u>uni-directional</u> if among all the chains from source <u>a</u> to sink <u>b</u> all chain flows occur in the same direction in any one arc.

2 DISCUSSION

It is often desirable to associate an abstract graph with a topological graph [1]. This can be done by representing each vertex (node) as a point in Euclidean three space and each arc as a curve joining its end vertices. In keeping with the definitions above, these curves do not intersect except at vertices. A graph is said to be planar if the associated topological graph is planar.[1] Whitney [1] gives other criteria for determining whether a graph is planar. These criteria are stated in the following two theorems.

Theorem 1. A graph is planar if, and only if, it has a dual

Theorem 2. A graph is planar if, and only if, it contains neither of the following two graphs as subgraphs:

(A) G' consisting of five nodes a, b, c, d, e, and arcs connecting each of these nodes to every other one by a single arc or a suspended chain. (Note. A suspended chain is a chain in which only the end vertices are met by more than two arcs.)

(B) G'' consisting of two sets of three nodes each a, b, c, and d, e, f and arcs connecting each of the nodes of the first set to every node of the second set by a single arc or a suspended chain.

1. A topological graph is planar if it can be mapped 1-1 onto a plane or a sphere.

Graph G'' is a representation of a classical problem in graph theory - the gas, water, electricity problem.

Representations.

An obvious representation of a graph is a diagram in which points are used to represent nodes and lines to represent arcs. Capacities can be shown as numbers adjacent to the arcs to which they apply. Orientation is usually denoted by arrows.

Another representation of a graph is a square matrix in which both the rows and the columns represent nodes. If we desire to represent the graph G by a matrix $L = \ell_{ij}$, then $\ell_{ij} = 1$ if the arc γ (ij) is in the graph and is zero otherwise. We could modify this matrix to show capacities by replacing the 1's with numbers which represent capacities. In an oriented graph ij may differ from ℓ_{ji} - in fact, one may be positive and the other zero.

Another matrix representation of a graph is an incidence matrix. In this matrix the nodes are represented by rows, the arcs by columns. We say an arc is incident on a node (and vice-versa) if the arc intersects the node. In an oriented (directed) graph we say that an arc which comes in to a node is positively incident on that node, an arc which goes out from a node is negatively incident on it. If we represent positive incidence by a +1, negative incidence by a -1, and no incidence by a zero, then in the incidence matrix of a directed graph each column would have one +1 and one -1. Rows which represent origins would contain only -1's. A row of exclusively +1's would indicate a terminal. Junction points would be represented by rows containing 1's of mixed signs.

BIBLIOGRAPHY

1. Whitney, H., "Non-Seperable and Planar Graphs", Transactions of the American Mathematical Society, Volume 34, pp. 339-362 (1932).